AD-A104 035 LOGICON INC SAN DIEGO CA F/G 5/9
AFT PROGRAM DESCRIPTION NAVIGATION/STRIKE TASKS, PHASE II,(U)
SEP 72 R M JOHNSON N61339-72-C-0108
UNCLASSIFIED SDR-120 NL

1 of 3
AD
A104 035

LEVEL II

135

1

ADA104035

AFT PROGRAM DESCRIPTION

NAVIGATION/STRIKE TASKS (PHASE II)

R. M. Jonnson

Contract No. 61339-72-C-0108

LOGICON Report SDR-120

Prepared for:

Naval Training Equipment Center
Orlando, Florida 32813

September 1972

AFT PROGRAM DESCRIPTION

NAVIGATION/STRIKE TASKS (PHASE II)

R. M. Johnson

Contract No. N61339-72-C-0108

LOGICON Report SDR-120

Prepared for:

Naval Training Equipment Center
Orlando, Florida 32813

September 1972

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS (Cont)

LIST OF TABLES

SECTION 1

INTRODUCTION

The Automated Flight Training (AFT) Program was designed as an ancillary program to the existing F-4 simulation and Automated Trainer Evaluation (ATE) programs used in the TRADEC System. The AFT Program, modular in design, operates primarily in the background mode of the F-4 program, and is a logical extension of the ATE program developed earlier. Detailed descriptions of each module, including flow charts, are presented in Section 2, and a brief description of each Meta-Symbol procedure employed in the AFT program is reviewed.

Each of the 28 separate modules described in Section 2 contain the information in the following order.

a. Program Module Name

b. Purpose

c. Requirements

d. Description

e. Inputs

f. Outputs

g. Program Entrances

h. Exits

i. Subroutines Called

j. Memory Requirements

k. Type of Program Module

l. Flow Charts.

LOGICON considers this depth of program description essential for NTEC to use the AFT Program effectively, and with confidence.

SECTION 2

MODULE DESCRIPTIONS

2.1 META-SYMBOL PROCEDURE

a. Program Name. ATE$PROC

b. Purpose. The purpose of the ATE$PROC module is to provide the Meta-Symbol assembler unique code sequences during assembly of the AFT Program.

c. Requirements. The ATE$PROC module is required to make available to each AFT Program module any defined procedure.

d. Description. Meta-Symbol Procedures (PROCs) are extensively described in the 'Symbol/Meta-Symbol Reference Manual.' In order to make the ATE$PROC module available to all AFT Program modules, it is necessary to place the ATE$PROC procedures on disc prior to assembly of the AFT Program modules. A card deck containing the ATE$PROC procedures to be executed under control of the Batch Processing Monitor (BPM) is provided. A listing of each ATE$PROC procedure can be found in the front of the MCC program listing.

The following PROCs are currently available to the AFT Program:

1. Procedure CMSG. This procedure, used to compose outgoing messages to the COGNITRONICS Speechmaker, has the following format:

   LABEL CMSG, P, CG, SN CWA(1), CWA(2), . . . , CWA(I), . . . , CWA(N)

   where:

   LABEL is the message name

   CMSG is the procedure name

   P is the message priority (1 through 10). Lowest numbers have the highest priorities.

CG is a message group number

SN is the message sequence number

CWA(I) are the COGNITRONICS word addresses

Example:

CMSGHBK CMSG, 2, 2 X'AE', X'96'
would generate the code corresponding
to the COGNITRONICS message:

HOLD BANK

and assign it a priority of 2, a message
group number of 2, and a sequence number
of 0 (omitted).

2. Procedure SAVREG. This procedure is used to save a sequence of general registers in the indicated memory locations. It has the following format:

LABEL SAVREG, R, N, LOC

where:

LABEL is the Meta-Symbol label, if any

SAVREG is the procedure name

R is the starting register number

N is the number of registers to be saved

LOC is the starting memory address where the registers are to be saved

3. Procedure RESREG. This procedure is used to restore a sequence of general registers from the indicated memory locations. It has the following format:

LABEL RESREG, R, N, LOC

where:

LABEL is the Meta-Symbol label, if any

RESREG is the procedure name

R is the starting register number

N is the number of registers to be restored

LOC is the starting memory address from which the registers are to be retrieved

4. Procedures ACT and DACT. These procedures are used to generate the calling sequence to activate or deactivate a program in either the AFT Foreground or Background Status Tables. Program numbers correspond to the sequence of the program in the appropriate status table list. Since this procedure employs a branch to either the CHGSTA:A or CHGSTA:D subroutine, it is necessary to reference (REF) these subroutines in the ATE Program modules as shown in the following example.

```
.
.
.
REF CHGSTA:A, CHGSTA:D
.
.
.
ACT GPM:1 ACTIVATE GENERAL PERFORMANCE
              MONITOR PROGRAM

DACT COG:1, IDI:1  DEACTIVATE COGNITRONICS AND
                   IDIIOM PROGRAMS
.
.
.
END
```

5. Procedures BIL and BOL. These procedures are pseudo-instructions for the Branch If Within Limits (BIL) and Branch If Out of Limits (BOL) instructions, customarily used after the Compare with Limits in Memory (CLM) instruction.

BIL generates a BCS, 6 instruction

BOL generates a BCR, 6 instruction

6. Procedure PUT. This procedure, used to transfer general registers to main memory, has the following format:

LABEL PUT, N, R, LOC, X

where

LABEL is the Meta-Symbol label, if any

PUT is the procedure name

N is the number of registers to be saved

R is the starting register number

LOC is the starting location in memory

X is the index value, if any

Example:

PUT, 3, 8, SAVE

Stores general register 8 into location SAVE

Stores general register 9 into location SAVE + 1

Stores general register 10 into location SAVE + 2

7. Procedure ASC. This procedure is used to convert an EBCDIC character set to ASCII characters for subsequent output to the IDIIOM display. It has the following format:

LABEL ASC X, X, X, X

where:

LABEL is the Meta-Symbol label, if any

ASC is the procedure name

X is the EBCDIC character string to be converted to an ASCII character string

Example:

ASC SP, PLUS, 1, SP would generate the ASCII character string ∧ + 1∧ (∧ indicates a blank character)

The following character set is currently included in this procedure.

EBCDIC	ASCII	EBCDIC	ASCII	EBCDIC	ASCII
SP	∧	A	A	O	O
PLUS	+	B	B	P	P
MINUS	-	C	C	Q	Q
SLASH	/	D	D	R	R
0	0	E	E	S	S
1	1	F	F	T	T
2	2	G	G	U	U
3	3	H	H	V	V
4	4	I	I	W	W
5	5	J	J	X	X
6	6	K	K	Y	Y
7	7	L	L	Z	Z
8	8	M	M		
9	9	N	N		

e. Inputs. None

f. Outputs. None

g. Program Entrances. Not applicable

h. Exits. Not applicable

i. Subroutines Called. None

j. Memory Requirements. None. Executed at assembly time

k. Type of Program Module. Meta-Symbol Procedures

l. Flow Charts. None

2.2 MODE AND CYCLE CONTROL (MODIFIED)

a. Program Module Name. Mode and Cycle Control (MCC) Modified

b. Purpose. The purpose of the MCC module for the F-4 simulation remains basically the same as that described in the F-4 documentation. In addition, it has been modified to provide for a flexible interface between the F-4 simulation and the ATE/AFT programs.

c. Requirements. In addition to the F-4 simulation, the following ATE/AFT requirements must be met:

1. Provide for real-time processing (50-millisecond program cycle) of the ATE/AFT Foreground Program modules.

2. Provide for processing of ATE/AFT Background program modules on a time available basis.

3. Maintain proper communications linkages between the F-4 and ATE/AFT programs.

4. Process the Sigma-7 Supervisory Console interrupt.

5. Process the COGNITRONICS interrupt.

6. Process the additional error codes supplied by the ATE/AFT programs.

7. Provide for JOYSTICK override control when requested from the monitor console.

d. Description. The ATE/AFT Foreground and Background programs are controlled by the ATE$F and ATE$B executive routines, respectively. Entries to the ATE$F executive routine were inserted near the ends of the MCC program cycles (NORMAL, CRASH/FREEZE, ZERO, and INSTRUMENT TEST). The ATE$F routine is executed just after the main F-4 simulation and just prior to the F-4 real time input/output. The ATE$B routine is processed after the normal F-4 program cycle (which now includes ATE$F) on a time available basis. All of the ATE/AFT Background programs are processed before proceeding to the F-4 Background programs. The F-4/ATE/AFT programs execution priority is:

1. F-4/ATE/AFT Foreground programs (every 50 milliseconds).

2. ATE/AFT Background programs.

3. F-4 Background programs.

The COGNITRONICS interrupt is triggered at hexadecimal location 6E whenever the COGNITRONICS device is ready for the next output word. Although the MCC program module was altered to process this interrupt, the output word selection and message queuing are done in the COG program module (ATE/AFT Background mode). The COGNITRONICS interrupt processor in MCC merely picks up the next word for output, initiates the I/O, and tests to ensure the output was successful. Since two audio channels of output are currently provided, the processor also incorporates a counter (KCOG) to determine which COGNITRONICS channel is being processed. The command/data buffer for COGNITRONICS output has the following format:

CMSG01	0000C861	Selects discrete outputs 216-223
CMS WORD	DATA	COG output address (supplied by COG program module)
	0000C851	Strobe out, discretes 184-191
	00000001	Apparent ZERO level for reset
	0000C851	Reset strobe, discretes 184-191
	00000000	Apparent ONE level for strobe dry

The Supervisory Console interrupt is also provided for in the modified MCC program. It is triggered at hexadecimal location 5D. The interrupt is acknowledged by setting the KEYINSW flag for further processing by the ATEX and I/O programs.

A provision for overriding student control in the cockpit is provided in the JOYSTICK modifications. Control of the simulator can be achieved at the Monitor Console by selection of the appropriately marked discrete switches. When this occurs, the student inputs for stick, throttle, flaps, and landing gear control are ignored. The actual processing of JOYSTICK override commands is accomplished in the F-4 CIV program module (a modification performed by NTEC personnel.)

The following additional ATE/AFT error codes have been inserted into the MCC program module for processing by the Error routine:

Error Code	Console Display	ATE Program	Error
X'101'	02000101	PMSG:1	Keyboard Printer Output Error
X'102'	02000102	KEYIN:1	Keyboard Printer Input Error
X'103'	02000103	MCC$1	COGNITRONICS I/O Not Successful
X'104'	02000104	MCC$1	COGNITRONICS Not Operative
X'105'	02000105	COG:1	COGNITRONICS Address Out Of Limits
X'106'	02000106	LPMSG:1	Line Printer Output Fault
X'107'	02000107	EXSC:1	Illegal Exercise Requested
X'108'	02000108	IFMTS:1	Parameter Monitor Table Too Large
X'109'	02000109	IFMTS:1	COGNITRONICS Address Not In ASTAB Table
X'110'	02000110	ALOGIC:2	Illegal Task Number Requested

e. Inputs

1. Internal Inputs (additional for ATE/AFT):

ATEFLAG	ATE/AFT Background Program Status
CMSWORD	COGNITRONICS Output Address

2. External Inputs (additional for ATE/AFT): None

3. Constants (additional for ATE/AFT): None

f. Outputs

1. Internal Outputs (additional for ATE/AFT):

BGACT	Flag which indicates ATE/AFT Background program was activated by an interrupt.

FTZERO	Flag which indicates the first cycle through Zero Mode.
KCOG	COGNITRONICS Channel Counter
KEYINSW	Flag which indicates Supervisory Console interrupt

2. External Outputs (additional for ATE/AFT): None

g. Program Entrances (additional for ATE/AFT)

1. COGNITRONICS Interrupt	Hexadecimal location 6E
2. Supervisory Console Interrupt	Hexadecimal location 5D

h. Exits (additional for ATE/AFT). None

i. Subroutines Called.

1. CRASH$1 ATE/AFT Crash Recovery Subroutine

2. RESET$1 ATE/AFT Reset-to-Zero Recovery Subroutine

j. Memory Requirements

1. Instructions 542

2. Data 2

k. Flow Charts. See figure 2-1.

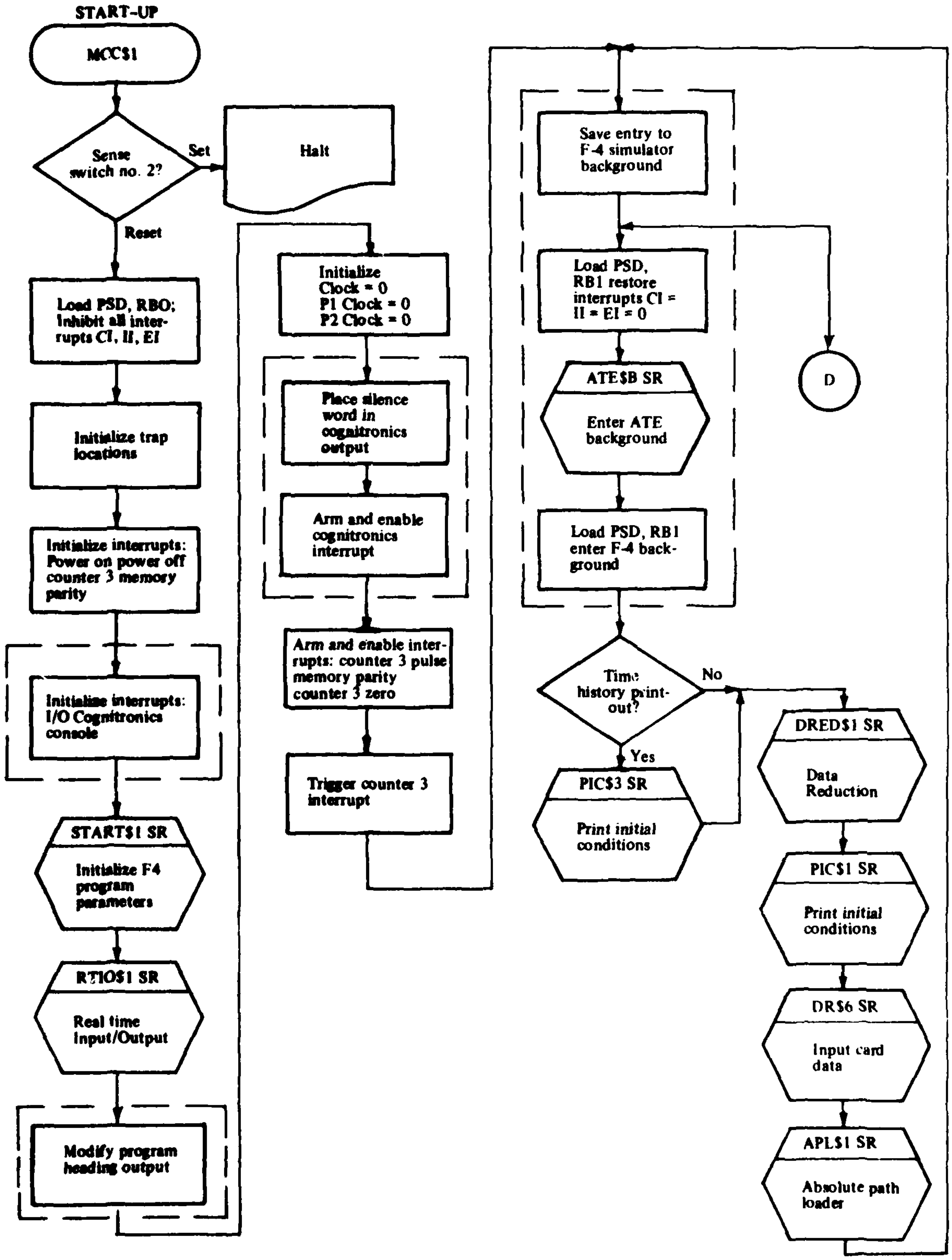


Figure 2-1. F-4 Simulator, Mode and Cycle Control Program (Sheet 1 of 4)

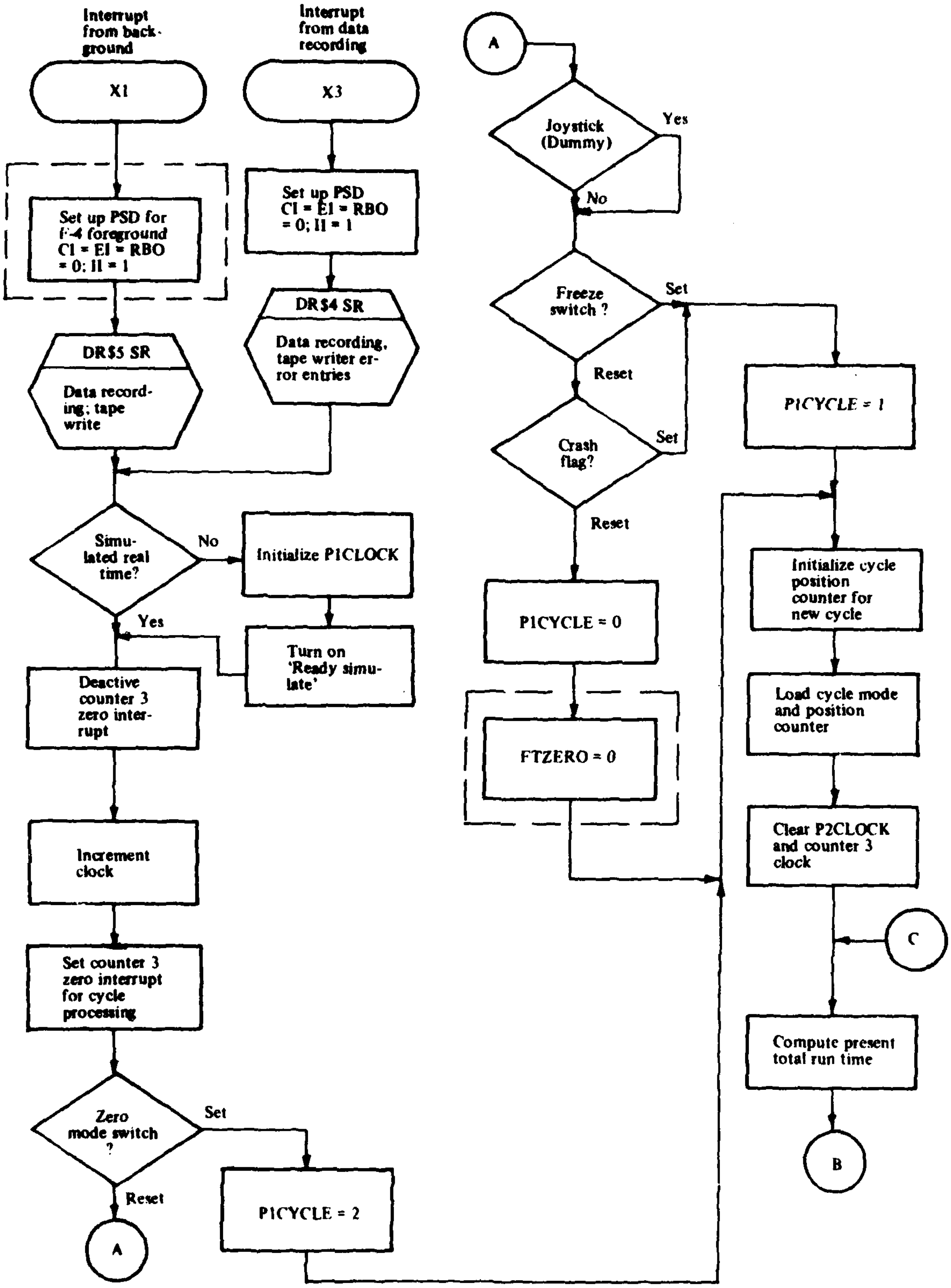
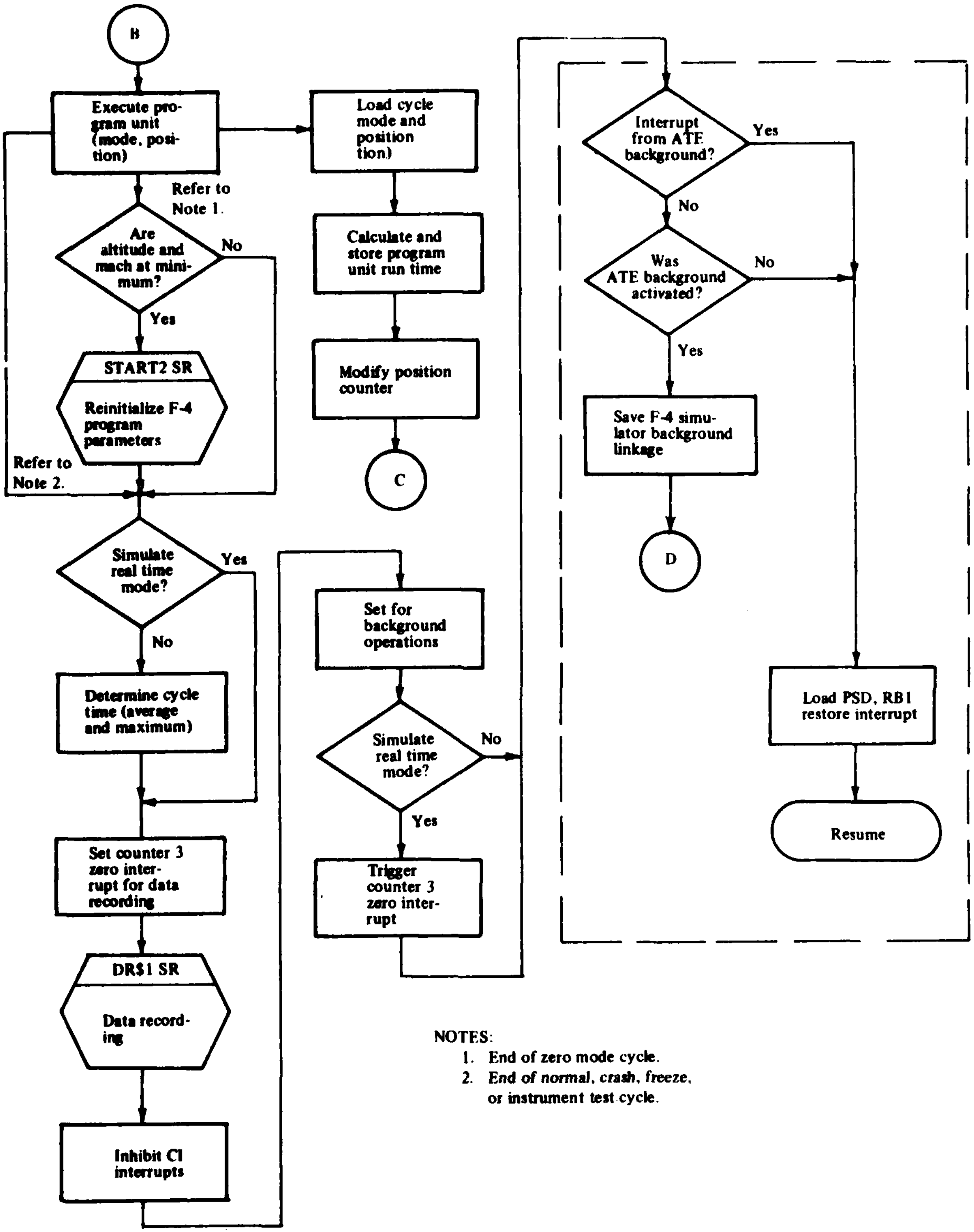


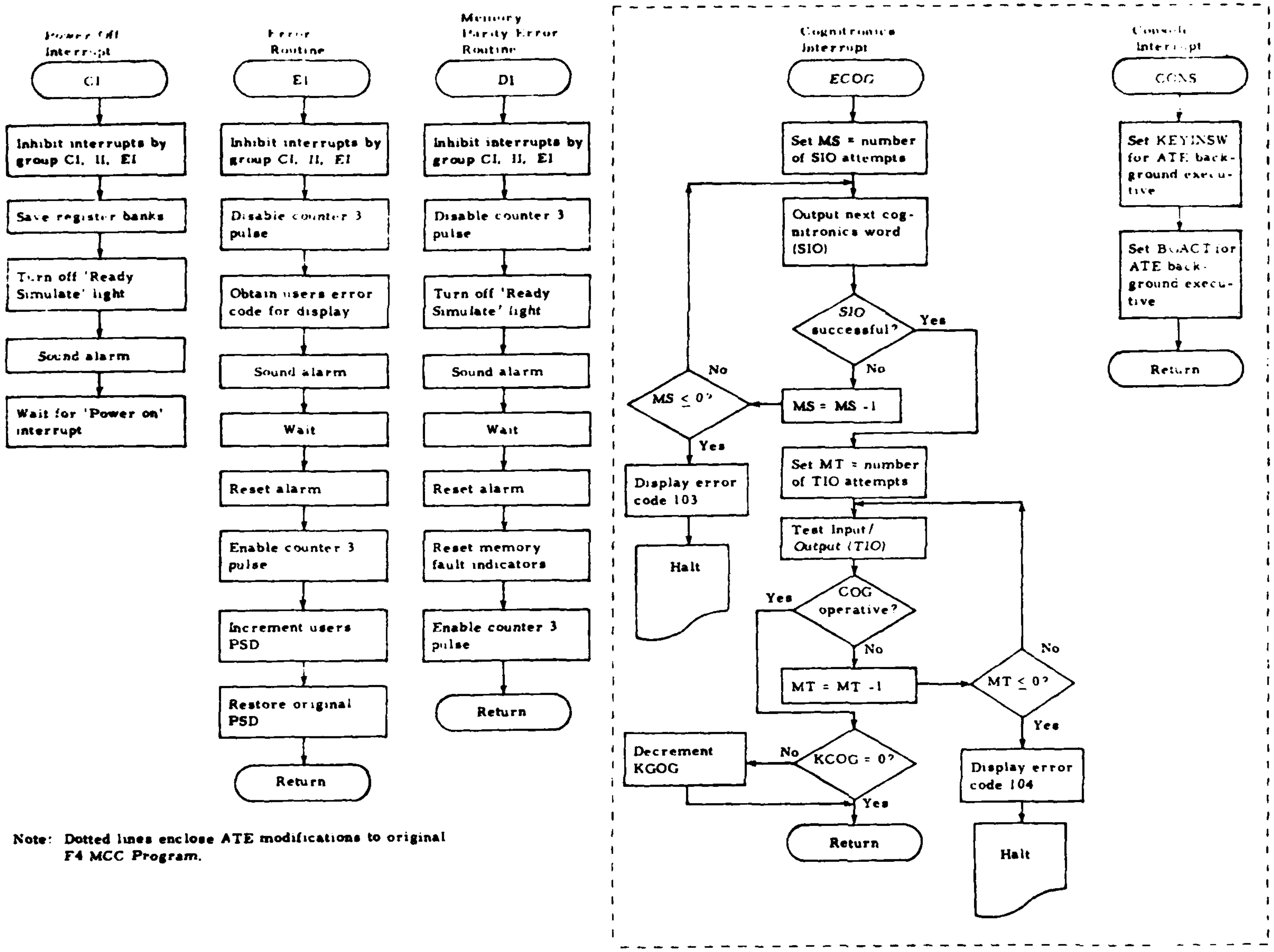
Figure 2-1. F-4 Simulator, Mode and Cycle Control Program (Sheet 2 of 4)



NOTES:

1. End of zero mode cycle.
2. End of normal, crash, freeze, or instrument test cycle.

Figure 2-1. F-4 Simulator, Mode and Cycle Control Program (Sheet 3 of 4)



Note: Dotted lines enclose ATE modifications to original F4 MCC Program.

Figure 2-1. F-4 Simulator, Mode and Cycle Control Program (Sheet 4 of 4)

2.3 ATE/AFT PARAMETERS

a. Program Module Name. ATE/AFT Parameters (APAM)

b. Purpose. The purpose of the APAM module is to provide a common storage area for all the parameters required for interprogram communications.

c. Requirements. The APAM module is required to:

1. Provide a common storage area for interprogram constants, variables, and other required data utilized in the ATE/AFT programs.

2. Define (DEF) specified symbols in order to make them available for all current or future training programs.

d. Description. In order to provide for efficient interprogram communications, the constants, variables, and other program data are placed in a common nonexecutable program module designated APAM.

e. Inputs. None

f. Outputs. None

g. Program Entrances. None

h. Exits. None

i. Subroutines Called. None

j. Memory Requirements

1. Instructions 0

2. Data 2052

k. Type of Program Module. Data

l. Flow Charts. None

2.4 NAVIGATION/STRIKE SYSTEM PARAMETERS

a. Program Module Name. Navigation/Strike System Parameters (NPAM)

b. Purpose. The purpose of the NPAM module is to provide the data and tables required to define each Navigation/Strike (NST) task, run (leg), and the parameters to be utilized for these maneuvers.

c. Requirements. The NPAM module is required to:

1. Describe each task employed in the NST maneuvers in terms of run time, parameters to be monitored, initial conditions, terminating conditions, etc.

2. Describe in detail the requirements of each parameter for each task defined in item 1.

d. Description. The NST task sequence of execution is specified in tables CLTDTL, ATTDTL, and DETDTL for the climb, attack, and descent tasks, respectively. These tables contain the task number and a pointer to the appropriate task description table (TDT) contained in the Task Description Parameter (TDP) module. The task description tables are composed of a variable list of data which describe, in detail, the task to be executed. TDT parameters have the following one-word format:

0 — 8	9 — 32
N	PAM

where:

N is a number defining the disposition of the value PAM.

PAM is a value or data pointer whose usage depends upon the action described by N.

Currently, the following values of N are employed:

N	ACTION TO BE TAKEN
0	End of Task Description Table (TDT). If PAM is zero, no further legs are associated with this task. If PAM is nonzero, the value contained in PAM is a pointer to another TDT which describes the next leg of this task.

1 PAM contains run time in 50-millisecond counts.

2 PAM contains a pointer to a Parameter Description Table which describes the requirements of the indicated parameter. (Parameter Description Tables are defined later in this paragraph).

3 PAM is a pointer to the task briefing message associated with this task.

4 PAM is a pointer to a memory location which contains the required T99DO3 bit configuration that must be attained by the flight program prior to the start of the run. T99DO3 represents the aircraft status with respect to landing gear, flaps, engine conditions, etc.

5 PAM points to a table which describes certain flight parameters and the limits they must be within before the run is initiated.

6 PAM points to a table which describes certain flight parameters and the limits they must fall within before the run can be terminated.

7 PAM points to the initial heading value for the run.

8 PAM points to the initial altitude value for the run.

9 PAM points to the initial airspeed or mach value for the run.

10 PAM points to the difficulty factors for the next leg, if another leg follows and the difficulty factors are different from those of the first leg.

11 PAM points to the display buffers to be used for the data transfer of heading and altitude.

12 PAM points to the display buffer to be used for the data transfer of airspeed or mach.

13 PAM points to the initial target bearing data for the selected attack task.

14 PAM points to the initial target range and turn data for the selected attack task.

15 This value is used for any climb or descent run where the course is to be reversed at a specified point. PAM points to the location containing the reciprocal course.

Interpretation of the Task Description Tables is done in the Task Selector (NSTS) program module.

One example of a typical task description table is as follows:

```
ATTDT10    RES     0
           PDEF    1, TA09
           PDEF    2, PSIATT
           PDEF    2, ALT20K
           PDEF    2, VGP9
           PDEF    2, PHI45
           PDEF    2, RI45
           PDEF    2, PA01
           PDEF    2, BT01
           PDEF    2, ALF01
           PDEF    2, RC01
           PDEF    3, ABMSG
           PDEF    5, ASPAM
           PDEF    7, IPS04A
           PDEF    13, TB10
           PDEF    14,  TR10
           PDEF    0, 0
```

The foregoing TDT represents a starboard quarter-to-stern (120 degrees) attack task. Target heading is 360 degrees, altitude 20,000 feet, and airspeed 0.9 mach. Initial interceptor conditions are:

Heading: 240 degrees
Altitude: 20,000 feet
Airspeed: 0.9 mach
Target relative bearing: 5 degrees
Target range: 7 miles
Approximate bank angle
for intercept: 45 degrees starboard

Tables 2-1, 2-2, and 2-3 list the parameters used for the climb, attack, and descent tasks, respectively. Tables 2-4 and 2-5 list the configuration and turbulence values, respectively.

Table 2-1. Climb Task Parameter Values

Task Number	Leg	Programmed Instructions	Initial Conditions	Terminating Conditions	Configuration*	Turbulence*
CL01 CL02 CL03 CL04 CL05 CL06	A	Climb to 20 Thousand Feet on Heading 045 Degrees at 100 Percent Power and 350 Knots.	1. Airborne 2. Heading 045°	1. 20,000' 2. 045° 3. 350 kts.	1 1 2 2 3 3	1 2 2 3 2 3
CL07 CL08 CL09 CL10 CL11 CL12	A	Climb to 15 Hundred Feet, Heading 045 Degrees, Speed 280 Knots.	1. Airborne 2. Heading 045°	1. 1,500' 2. 045° 3. 280 kts.	1 1 2 2 3 3	1 2 2 3 2 3
	B	Reverse Course 180, Climb to 20 Thousand Feet at 100 Percent Power and 350 Knots.	1. 1,500' 2. 045° 3. 280 kts.	1. 20,000' 2. 225° 3. 350 kts.		
CL13 CL14 CL15 CL16 CL17 CL18	A	Climb to 15 Hundred Feet, Heading 360 Degrees, Speed 350 knots.	1. Airborne 2. Heading 045°	1. 1,500' 2. 360° 3. 350 kts.	2 2 2 3 3 3	1 2 3 1 2 3
	B	Vector Port 330, Climbing To 20 Thousand Feet At 100 Percent Power And Mach Point 9.	1. 1,500' 2. 360°	1. 10,000' 2. 330° 3. .9 mach		
	C	Vector Starboard 060.	1. 10,000' 2. 330° 3. .9 mach	1. 20,000' 2. 060° 3. .9 mach		

*Refer to tables 2-4 and 2-5 for configuration and turbulence values

Table 2-2. Attack Task Parameter Values

Task Number	Initial Attack Heading (deg.)	Relative Target Bearing	Target Range (n. m.)	Initial Bank	Nominal Run Time (sec.)	Initial Distance to Right of Target (n. m.)	Initial Distance Behind Target (n. m.)	Configuration*	Turbulence*
AT01	090	23^{o} stbd.	8.0	30^{o} port	77.2	-7.4	-3.2	1	1
AT02	270	23^{o} port	8.0	30^{o} stbd.	77.2	+7.4	-3.2	1	2
AT03	120	6^{o} stbd.	13.7	30^{o} port	103.0	-11.1	-8.1	1	3
AT04	240	6^{o} port	13.7	30^{o} stbd.	103.0	+11.1	-8.1	2	1
AT05	180	34^{o} port	26.7	30^{o} port	154.4	-14.8	-22.3	2	2
AT06	180	34^{o} stbd.	26.7	30^{o} stbd.	154.4	+14.8	-22.3	2	3
AT07	090	19^{o} stbd.	4.5	45^{o} port	44.6	-4.3	-1.4	1	1
AT08	270	19^{o} port	4.5	45^{o} stbd.	44.6	+4.3	-1.4	1	2
AT09	120	5^{o} stbd.	7.7	45^{o} port	59.5	-6.4	-4.3	2	2

NOTE (1): All runs use target heading 360^{o}, speed .9 mach, altitude 20,000'
NOTE (2): Intercept data is based upon interceptor speed = .9 mach, 20,000' altitude
*See tables 2-4 and 2-5 for configuration and turbulence values

Table 2-2. Attack Task Parameter Values (Cont)

Task Number	Initial Attack Heading (deg.)	Relative Target Bearing	Target Range (n.m.)	Initial Bank	Nominal Run Time (sec.)	Initial Distance to Right of Target (n.m.)	Initial Distance Behind Target (n.m.)	Configuration*	Turbulence*
AT10	240	5^{o} port	7.7	45^{o} stbd.	59.5	+6.4	-4.3	2	3
AT11	180	35^{o} port	15.1	45^{o} port	89.2	-8.6	-12.4	3	2
AT12	180	35^{o} stbd.	15.1	45^{o} stbd.	89.2	+8.6	-12.4	3	3
AT13	090	9^{o} stbd.	2.5	60^{o} port	25.7	-2.5	-0.4	2	1
AT14	270	9^{o} port	2.5	60^{o} stbd.	25.7	+2.5	-0.4	2	2
AT15	120	ahead	4.2	60^{o} port	34.3	-3.7	-2.0	2	3
AT16	240	ahead	4.2	60^{o} stbd.	34.3	+3.7	-2.0	3	1
AT17	180	36^{o} port	8.4	60^{o} port	51.5	-5.0	-6.8	3	2
AT18	180	36^{o} stbd.	8.4	60^{o} stbd.	51.5	+5.0	-6.8	3	3

NOTE (1): All runs use target heading 360^{o}, speed .9 mach, altitude 20,000'
NOTE (2): Intercept data is based upon interceptor speed = .9 mach, 20,000' altitude
See tables 2-4 and 2-5 for configuration and turbulence values

Table 2-3. Descent Task Parameter Values

Task Number	Leg	Programmed Instructions	Initial Conditions	Terminating Conditions	Configuration*	Turbulence*
DE01 DE02 DE03 DE04 DE05 DE06	A	Power Back, Begin Descent to 15 Hundred Feet, Maintain Speed 250 Knots.	1. 20,000' 2. Present Heading	1. 1,500' 2. Present Heading 3. 250 Knots	1 1 2 2 3 3	1 2 2 3 2 3
DE07 DE08 DE09 DE10 DE11 DE12	A	Power 70 Percent, Speed Brake out, Begin Descent Maintaining Speed Point 8.	1. 20,000' 2. Present Heading	1. 10,000' 2. Present Heading 3. .8 mach	1 1 2 2 3 3	1 2 2 3 2 3
	B	Reverse Course 180, Level Off at 15 Hundred Feet, Speed 250 Knots.	1. 10,000' 2. Present Heading 3. .8 Mach	1. 1,500' 2. Reciprocal Heading 3. 250 Knots		
DE13 DE14 DE15 DE16 DE17 DE18	A	Steer Heading 330, Maintain 20 Thousand, Reduce Power, Hold Speed 280 Knots.	1. 20,000' 2. Present Heading	1. 20,000' 2. 330° 3. 280 Knots	2 2 2 3 3 3	1 2 3 1 2 3
	B	Turn Port Heading 270 Degrees, Power 70 Percent, Speed Brake Out, Begin Descent at 300 Knots.	1. 20,000' 2. 330° 3. 280 Knots	1. 10,000' 2. 270° 3. 300 Knots		
	C	Turn Port Heading 045 Degrees at Thirty Degrees Bank, Level Off at 15 Hundred Feet, Maintain Heading and Slow to 250 Knots.	1. 10,000' 2. 270° 3. 300 Knots	1. 1,500' 2. 045° 3. 250 Knots		

*See tables 2-4 and 2-5 for configuration and turbulence values

Table 2-4. Configuration Values (Adaptive Variables)

Configuration Value	Aircraft Weight (lb)	Internal Fuel (lb)	2 Wing Tanks (lb)	Center Tank (lb)	Sidewinder* Missiles (lb)	Gross Weight (lb)
1	31,100	2,400	–	–	–	33,500
2	31,100	2,400	5,674	–	–	39,174
3	31,100	2,400	5,674	4,200	1,204	44,578

*During attack runs, Sparrow missiles are also attached but are fired when trigger is depressed. This adds an additional 910 lb to gross weight.

Table 2-5. Turbulence Values (Adaptive Variables)

Turbulence Value	Percent of Maximum Value
1	0% (No Turbulence)
2	6% (Moderate Turbulence)
3	12% (Heavy Turbulence)

The pointers associated with N = 2 values are placed in the Parameter Monitor Table (PMTAB) for each run. Each pointer contains the address of the appropriate Parameter Description Table (PDT) to be employed in the run. PDTs describe the flight parameter's status, limits, etc., as well as reserving storage for parameter data to be accumulated during the run. Figure 2-2 depicts the PDT format.

Referring to Figure 2-2, the first halfword of word 0 describes the parameter status which has the following attributes:

1. If bit 0 = 1, the parameter is to be checked. Checking means that a count is to be kept of the number of times the parameter falls within the inner, middle, and outer limit categories. Words 13 through 15 are reserved for these count values.

2. If bit 1 = 1, the parameter is to be sampled. Sampling requires the accumulation of the error between the nominal value and actual value every 500 milliseconds. This error is normalized, squared,

	0 — 15	16 — 31
Word 0	STATUS	WA (PAM)
Word 1	PARAMETER NOMINAL VALUE (PC)	
Word 2	NUMBER OF SAMPLES (N)	
Word 3	SUM OF THE SQUARES - $\sum \frac{\lvert PC-PA \rvert^2}{PV}$	
Word 4	NORMALIZATION VALUE (PV)	
Word 5	PARAMETER INNER LIMIT (DPI)	
Word 6	PARAMETER MIDDLE LIMIT (DPM)	
Word 7	PARAMETER OUTER LIMIT (DPO)	
Word 8	WA (+ DPO MESSAGE)	WA (- DPO MESSAGE)
Word 9	WA (+ DPM MESSAGE)	WA (- DPM MESSAGE)
Word 10	WA (DPI MESSAGE)	WA (DISPLAY BUFFER)
Word 11	DISPLAY SCALE FACTOR	
Word 12	WA (LINE PRINTER TITLE)	WA (DISPLAY NOMENCLATURE)
Word 13	INNER LIMIT COUNT	
Word 14	MIDDLE LIMIT COUNT	
Word 15	OUTER LIMIT COUNT	

Figure 2-2. Parameter Description Table Format

summed with previous values, and retained in word 3 of the PDT. Word 2 of the PDT contains the number of samples accumulated.

3. Bit 2 is used to indicate that the actual parameter value is to be displayed. If this bit = 0, the difference between the actual value and nominal value will be displayed.

4. If bit 3 = 1, the parameter is to be displayed on the IDIIOM graphics display.

5. If bit 4 = 1, the parameter is expressed in degrees. The program is to provide a means of expressing the value modulo 360.

6. Bit 5 is currently unused.

7. If bit 6 = 1, the parameter is to be monitored only. In this instance, the program checks only to ensure that the parameter remains within the outer limits. If the parameter exceeds this value, the run is terminated. This procedure differs from checking in that a count of the inner, middle, and outer limit categories is not retained.

8. If bit 7 = 1, the parameter is to be scored at the end of the run (Refer to the NSDP module description).

9. If bit 8 = 1, the parameter is a mach (vs. airspeed) value.

10. Bit 9 = 0 indicates that bits 10-11 contain the display buffer number. Bit 9 = 1 indicates bits 10-11 contain the limits for the strike index.

11. Bits 10-11 contain the display buffer number (this value is ignored if bit 3 = 0) or the strike index value.

12. Bits 12-15 contain the termination code to be supplied if the parameter exceeds the outer limits listed in word 7.

The second halfword of word 0 is a pointer to the parameter itself and is provided as an easy access to obtain the parameter value, when needed. Word 1 contains the nominal value for the parameter during the run/leg. This value, PC, is compared to the actual parameter value, PA, to determine the parameter error. Words 2 and 3 contain, respectively, the number of times the parameter was sampled and the sum of the squares of the absolute errors. Word 4 contains the error normalization value. Words 5, 6, and 7 contain the inner, middle, and outer limits, respectively, for the parameter. These values are expressed as an absolute difference which is added to and subtracted from the nominal value to obtain the actual parameter range. See Figure 2-3. Word 8 contains pointers to the COGNITRONICS diagnostic messages associated with the parameter. Diagnostic messages are issued whenever the parameter exceeds the outer limit values. If the limit is exceeded in a positive direction, the message address in the first halfword is used; if negative, the message address in the second halfword is used. Whenever a parameter exceeds the outer limits, the run is terminated at that point. Word 9 is similar to word 8 except that it contains the message addresses for the COGNITRONICS instruction messages associated with the parameter. Instruction messages are issued whenever the parameter is beyond the middle limits but within the outer limits.

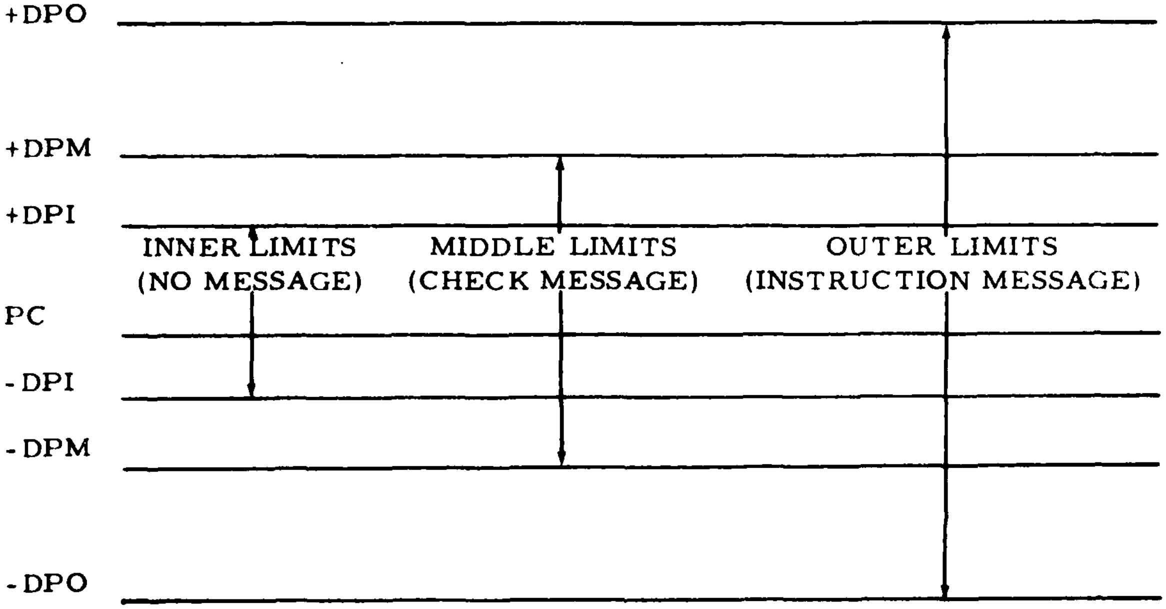


Figure 2-3. Diagram of Nominal Value PC With Limits

The first halfword of word 10 contains the message address of the COGNITRONICS check message associated with the parameter. A check message is transmitted whenever the parameter is beyond the inner limits but within the middle limits. If the parameter is to be displayed, the second halfword of word 10 contains the address of the display buffer associated with the parameter and word 11 contains the scale factor to be applied to the parameter before it is placed in the display buffer. The first halfword of word 12 is a pointer to the parameter title used for line printer output. The second halfword is a pointer to the nomenclature associated with the parameter for the display. Words 13, 14, and 15 contain the number of counts each time the parameter was detected in the inner, middle, and outer limit categories, respectively.

e. Inputs. None

f. Outputs. None

g. Program Entrances. None

h. Exits. None

i. Subroutines Called. None

j. Memory Requirements.

    1. Instructions 0

    2. Data 2391

k. Type of Program Module. Data

l. Flow Charts. None

2.5 ATE/AFT MODIFICATIONS TO F-4 PARAMETERS

a. Program Module Name. ATE/AFT Modifications to F-4 Simulation Parameters (AMOD)

b. Purpose. The purpose of the AMOD module is to modify specified F-4 simulator input parameters.

c. Requirements. The AMOD module is required to:

1. Overwrite the following F-4 system parameters with the values supplied by the ATE Adaptive Logic (ALOGIC:1) program, the AFT Adaptive Logic (ALOGIC:2) program, and the Nav/Strike Adaptive Logic (NSAL) subroutine.

DRA	Rough Air Input from the Monitor Console
DCT	Center Fuel Tank ON/OFF
DWT	Wing Fuel Tanks ON/OFF
DM	Sparrow Missiles Attached
DSW	Sidewinder Missiles Attached

2. To provide engine flame-out data to the Emergency Processor Routine (EMRP), when required.

d. Description. In order to implement the selected difficulty factors, it was necessary to modify certain F-4 system parameters that are normally controlled from the Monitor Console and the F-4 cockpit. Rather than make extensive changes to the existing F-4 program modules, the AMOD routine was designed to overwrite these parameters, subsequent to their input via the F-4 CIV routine but prior to their use by the other F-4 program modules. This technique, in effect, prohibits input of the following parameters from the monitor console.

1. Rough Air

2. Fuel Increment/Decrement

3. Center Tank

4. Wing Tanks

5. Sidewinder Missiles

6. Sparrow Missiles

The values to be supplied for these functions are computed and selected in the ATE Adaptive Logic (ALOGIC:1) program, AFT Adaptive Logic (ALOGIC:2) program, and Nav/Strike Adaptive Logic (NSAL) subroutine.

In addition, if the Emergency Procedures Exercise is being run, engine Master Switch positions are provided to allow the apparent flame-out of either engine when so directed by the ATE Emergency Procedures (EMRP) program.

e. Inputs

1. Internal Inputs:

ATECT	Center Tank Designator (1 = attached)
ATEM	Sparrow Missile Designator (0 = attached)
ATERAF	Rough Air Input
ATESW	Sidewinder Missiles Designator (1 = attached)
ATEWT	Wing Tanks Designator (1 = attached)
FOLE	Flame-out Left Engine Designator
FORE	Flame-out Right Engine Designator

2. External Inputs:

KKI	A/C On Ground Flag
RANDOM	Random Number
T99DI3	Discrete Input Word #3
T99DO3	Discrete Output Word #3
T99DO5	Discrete Output Word #5

3. Constants: None

f. Outputs

1. Internal Outputs: None

2. External Outputs:

DCT	F-4 Center Tank
DM	F-4 Sparrow Missiles
DRA	F-4 Rough Air
DSW	F-4 Sidewinder Missiles
DWT	F-4 Wing Tanks
T99DI3	Discrete Input Word #3 (as modified)
T99DO3	Discrete Output Word #3 (as modified)
T99DO5	Discrete Output Word #5 (as modified)
WCTF	Center Tank Fuel Quantity
WWTF	Wing Tanks Fuel Quantity

g. Program Entrances. BAL, 15 ATEMOD$1

h. Exits. B *ATEMEX (calling location +1)

i. Subroutines Called. None

j. Memory Requirements

1. Instructions 62

2. Data 3

k. Type of Program Module. Foreground Program

l. Flow Charts. See figure 2-4.

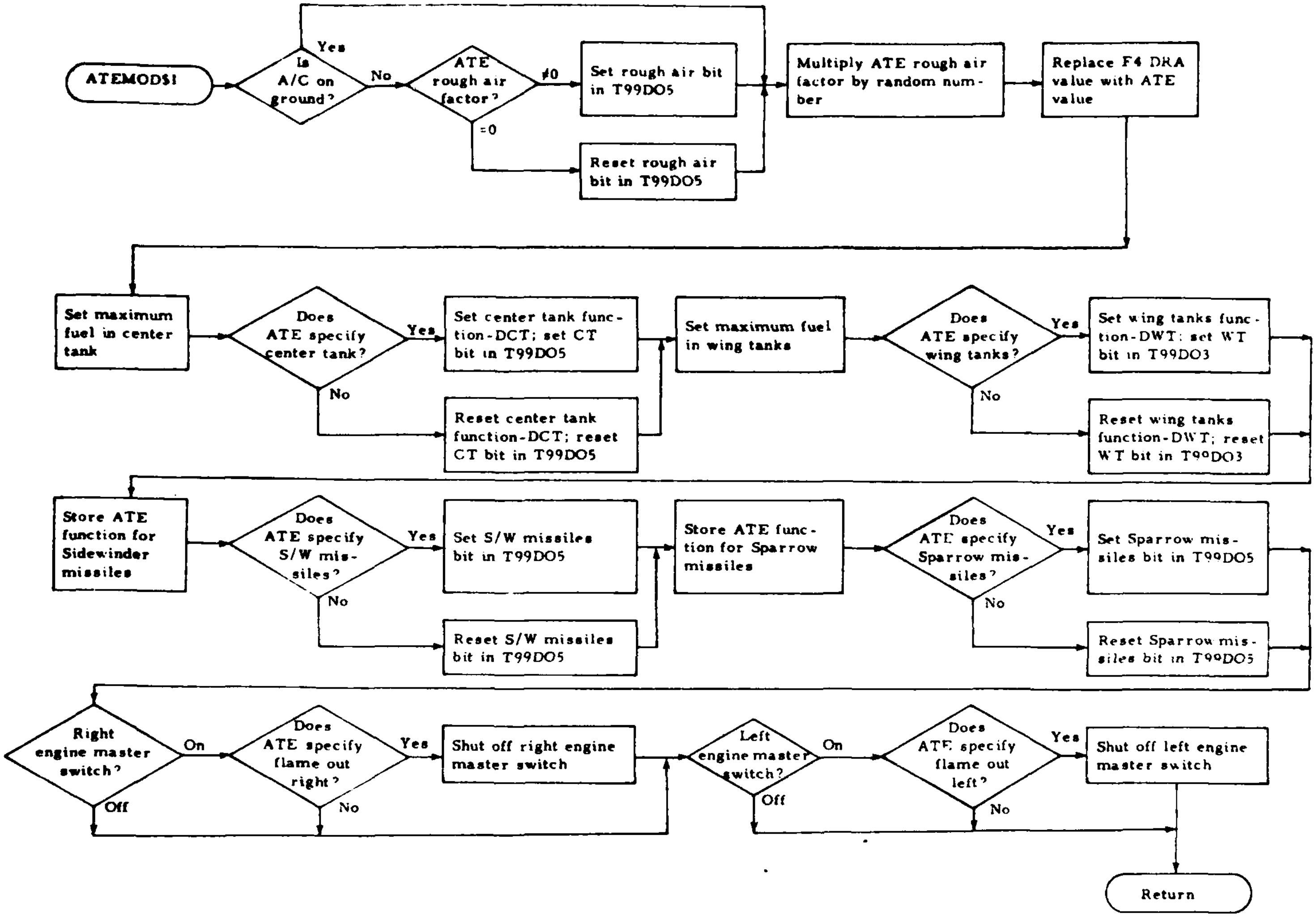


Figure 2-4. ATE/AFT Modifications to F-4 Simulation Parameters

2.6 AFT MODIFICATIONS TO F-4 PARAMETERS

a. Program Module Name. AFT Modifications to F-4 Simulation Parameters (AFTM)

b. Purpose. The purpose of the AFTM module is to modify the F-4E simulator parameters necessary to control the Instrument Flight Maneuvers and the Nav/Strike Maneuvers.

c. Requirements. The AFTM module is required to:

1. Override the speed brake control during IFM maneuvers.

2. Provide a program control for the simulator FREEZE condition when requested.

3. Compute the delta trim values for all three axes and determine if they are in limits.

d. Description. In order to implement the Instrument Flight Maneuvers, it is necessary to terminate control by the student pilot at specified times. This is done by imposing the FREEZE condition whenever the run is either successfully terminated or the student pilot exceeds the limits of the specified run. At this point, the FREEZE condition is initiated and the simulation proceeds under program control.

The Speed Brake switch is used in IFM by the student pilot to initiate the beginning of a run. Since the speed brake is used as a signal device for these maneuvers, the normal application of the speed brake no longer applies. The program was modified in such a way as to prevent speed brake manipulation from affecting the aerodynamic equations.

At specified places in the Instrument Flight Maneuvers and Nav/Strike Maneuvers, it is important to trim the aircraft to a desired condition prior to initiating the next run. The AFTM module computes the difference between actual trim and desired trim conditions about all three axes. The differences are then compared to limit values. Should any difference be out of limits, the appropriate trim control is activated to bring the actual trim within limits.

e. Inputs

1. Internal Inputs:

SBOV	Speed Brake Override
SBBIT	Speed Brake Condition Bits
ATEFZ	Program Control Freeze Condition
TRIMOVR	Trim Override Switch
TRIMACT	Trim Control Active Switch
LTRIM	Desired Longitudinal Trim
DSATRIM	Actual Aileron Trim
ATRIM	Desired Aileron Trim
DRPTRIM	Actual Rudder Trim
RTRIM	Desired Rudder Trim

2. External Inputs:

T99DI1	Discrete Input Word #1
T99DI3	Discrete Input Word #3
XT	Actual Longitudinal Trim

3. Constants:

SBBITC	Remove Speed Brake Bit Conditions
SBIN	Speed Brake In Bit Condition
FCONT3	Flight Control Conditions
LTLIM	Longitudinal Trim Limit
ATLIM	Aileron Trim Limit
RTLIM	Rudder Trim Limit

f. Outputs

1. Internal Outputs:

SBCON	Speed Brake Conditions
TRIMACT	Trim Active Switch

2. External Outputs:

T99DI1	Discrete Input Word #1
T99DI3	Discrete Input Word #3

g. Program Entrances. BAL, 15 ATEMOD$2

h. Exits. B *AMODEX (calling location + 1)

i. Subroutines Called. None

j. Memory Requirements

1. Instructions 63

2. Data 9

k. Type of Program Module. Foreground Program

l. Flow Charts. See figure 2-5.

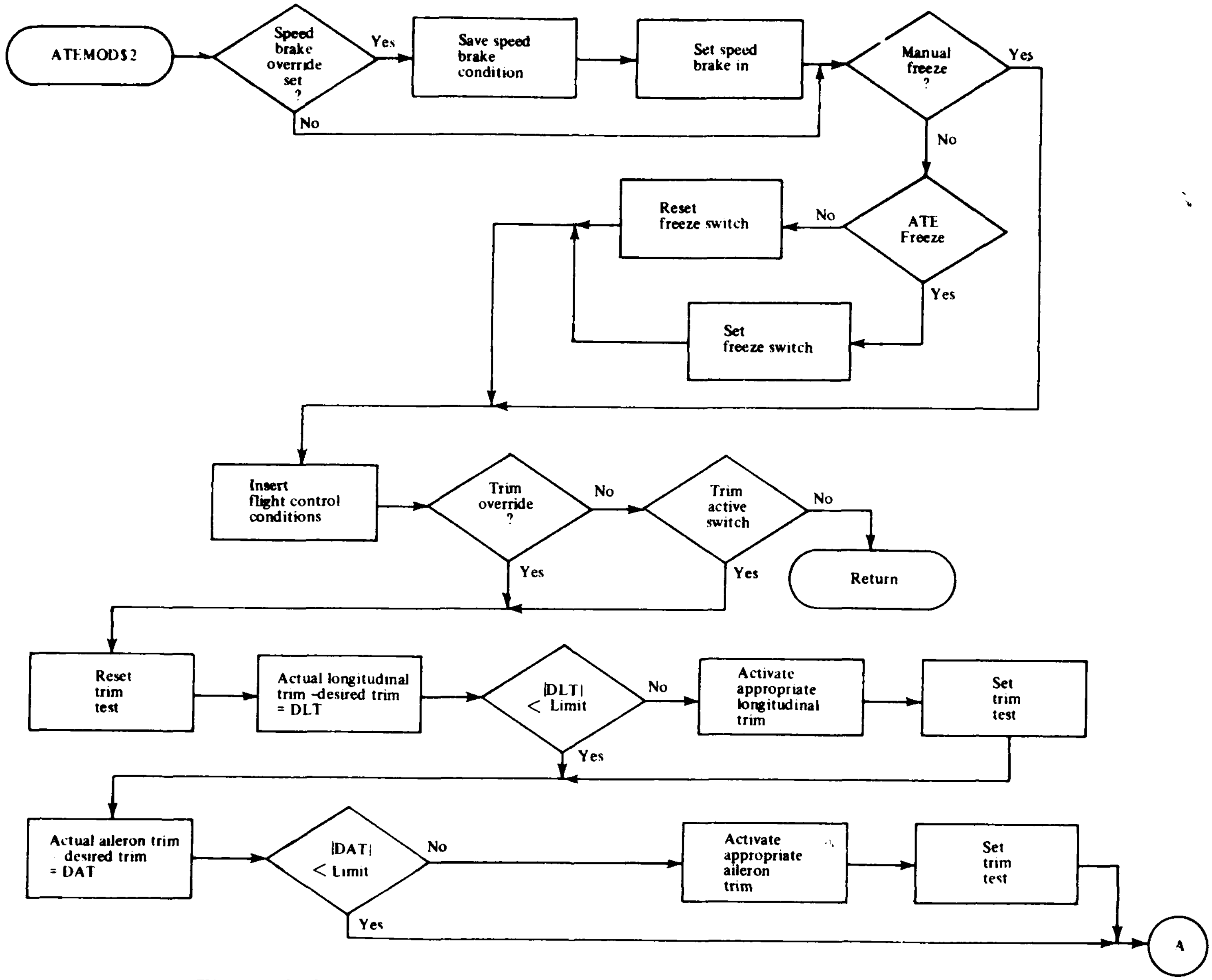


Figure 2-5. AFT Modifications to F-4 Parameters (Sheet 1 of 2)

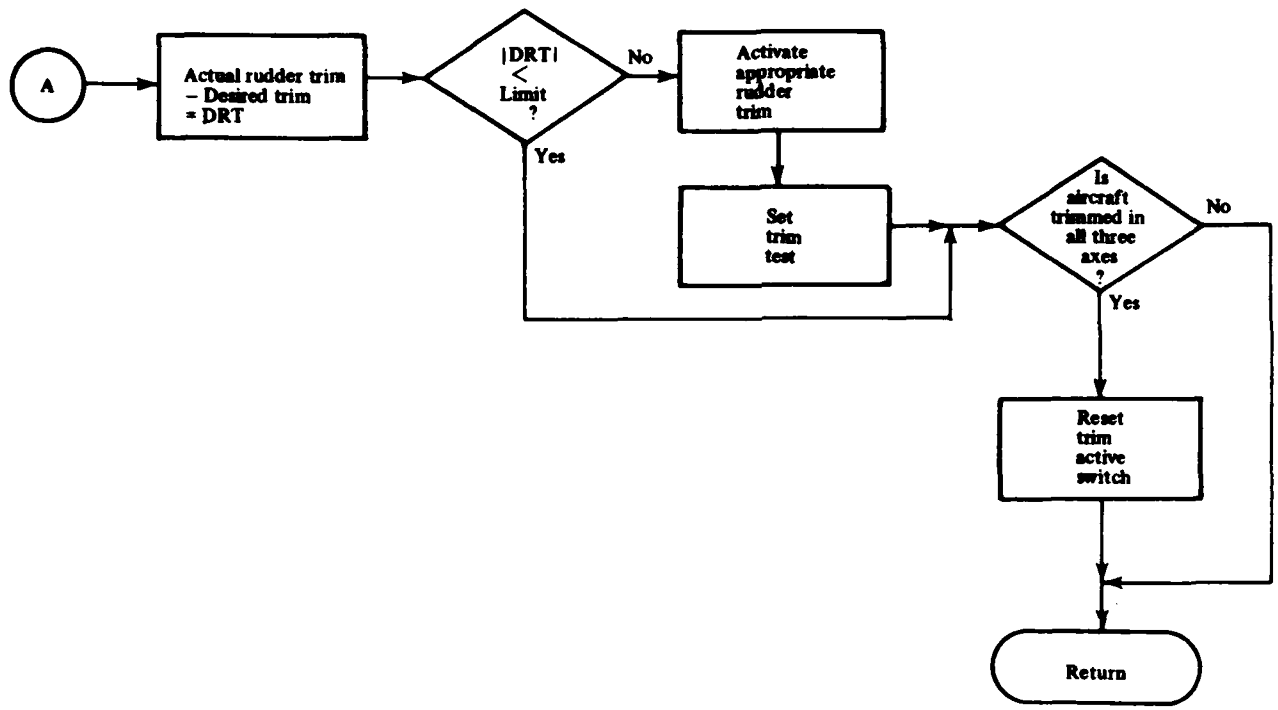


Figure 2-5. AFT Modifications to F-4 Parameters (Sheet 2 of 2)

2.7 ATE/AFT EXECUTIVE ROUTINES

a. Program Module Name. ATE/AFT Executive Routines (ATEX)

b. Purpose. The purpose of the ATEX program is to monitor and control program flow for the ATE/AFT Foreground and Background programs and to activate or deactivate designated programs in the Program Status Lists.

c. Requirements. The ATEX program must:

1. Search the ATE/AFT Foreground Program Status List every 50 milliseconds and execute each active program found.

2. Search the ATE/AFT Background Program Status List and execute each active program found.

3. Activate or deactivate programs contained in either the Foreground or Background Status Tables, when required.

4. Direct proper ATE/AFT program flow whenever the 'RESET-TO-ZERO' mode is selected.

5. Direct proper ATE/AFT program flow whenever the aircraft crashes.

6. Override the speed brake control, when requested.

7. Compute the average rate-of-climb and rate-of-turn parameters for program use.

8. Provide a program-controlled 'Reset-to-Zero' status, when required.

d. Description. Execution of all ATE/AFT programs (AMOD, AFTM, and subroutines excepted) is controlled by the ATE$F (Foreground Executive) routine or the ATE$B (Background Executive) routine. These routines interrogate the current status of the programs contained in the Program Status Lists and, if the program is active, it is executed. Programs within the Program Status Lists are arranged in order of priority — the highest priority being executed first. The format for Program Status Lists follows.

Foreground

FORETAB

0	1 ... 15	16 ... 31
A	(Spare)	PTR to Foreground, Program 1
A	(Spare)	PTR to Foreground, Program 2
A	(Spare)	PTR to Foreground, Program 3
⋮	⋮	⋮
A	(Spare)	PTR to Foreground, Program N
0	000	0 0 0 0

Background

BACKTAB

0	1 ... 15	16 ... 31
A	(Spare)	PTR to Background, Program 1
A	(Spare)	PTR to Background, Program 2
⋮	⋮	⋮
A	(Spare)	PTR to Background, Program N
0	000	0 0 0 0

Program List Word Status

NEGATIVE (Bit 0 = 1): Program Active (to be executed)

ZERO (Bits 0-31 = 0): End of Program List (return control to calling program)

POSITIVE (Bit 0 = 0): Program Inactive (do not execute)

The major features provided by this technique are:

1. Programs can be list ordered by execution priority.
2. Executive routines are completely independent of other program modules.
3. Any active program has the capability to activate or deactivate any other Foreground/Background program.
4. Priority of any Foreground/Background program can be readily changed by simply reordering the appropriate program list.
5. Inactive programs are readily bypassed.
6. New Foreground/Background programs are easily added by assembling the program module and inserting a one-word linkage in the list.
7. Obsolete Foreground/Background programs can be removed by removing the program module and deleting the one-word linkage from the list.
8. Foreground/Background programs can be virtually removed by simply deleting the one-word linkage from the list.
9. Foreground programs can be transferred to background mode (and vice versa) by interchanging the one-word linkage.
10. Tables are open-ended and may be of any length.

Two Sigma-7 Meta-symbol Procedures (PROCs) are employed to either activate or deactivate ATE programs listed in the Program Status Tables. PROC ACT activates and PROC DACT deactivates the programs contained in the corresponding address field. Any number of programs may be activated or deactivated with either one of these procedures as indicated in the following examples:

```
ACT     NSST:1

DACT    TPM:1, SDM:1
```

In the preceding examples, procedure ACT activates the Attack Monitor Program module and procedure DACT deactivates the Task Performance Monitor and Steering Dot Monitor modules.

The ACT and DACT PROCs, along with program module names, are listed in the front section of the MCC Program Listing. PROC Execution is accomplished by branching to either CHGSTA:A (ACT) or CHGSTA:D (DACT) for each program to be activated or deactivated. Whenever the RESET-TO-ZERO button is depressed on the Monitor Console, the RESET$1 routine is executed. This routine resets program parameters by deactivating the programs in the Foreground/Background Status Lists, resetting the GO flag (thus allowing Student File Input commands), purging the COGNITRONICS message queue, and providing for silence output to the COGNITRONICS Speechmaker.

Whenever the F-4 program detects an aircraft CRASH condition, the CRASH$1 routine is executed. This routine also resets program parameters by clearing the Status Lists, purging the COGNITRONICS Speechmaker, and activating the Exercise Termination (EXTR) program.

The rate-of-turn and rate-of-climb values provided in the F-4 program are instantaneous and must be smoothed before efficient use by the IFM programs. This function is performed by the ATE$F program. A moving average technique is performed where the F-4, 50-millisecond values are averaged over a half-second period and the last four half-second values are averaged over a 2-second period.

e. <u>Inputs</u>

1. Internal Inputs:

BGACT	Background Program Interrupt Flag
CEXER	Current Exercise Number
COSPHI	Cosine of Bank Angle
COSPSIF4	Cosine of Heading
FTZERO	1st Cycle through ZERO Mode
KPSI	Pseudo-Heading Sample Counter
PSIS	Pseudo-Heading
PSIM1	Pseudo-Heading-Last Cycle
KEYINSW	Supervisory Console Interrupt Flag
SBLAST	Speed Brake Control Status—Last Cycle
SBCON	Speed Brake Control Status

SBOVR	Speed Brake Override Switch
SCSW	Session Complete Switch
SBPOS	Actual Speed Brake Position
SINPHI	Sine of Bank Angle
SINPSIF4	Sine of Heading

2. External Inputs:

CRASH	F-4 Crash Flag
FREEZE	F-4 Freeze Flag
RIC	Indicated Rate-of-Climb
PA	Roll Rate (radians/second)
PSIB	Base Heading
PSIUP	Heading Update Switch

3. Constants:

BACKTAB	Background Program Status List
FORETAB	Foreground Program Status List
SBIN	Speed brake IN condition
SBOUT	Speed brake OUT condition

f. <u>Outputs</u>

1. Internal Outputs:

ATEFLAG	Background Program Status Flag
ATEFZ	Program Controlled FREEZE Switch
BGACT	Background Program Interrupt Flag
CMHADDR	Address of COGNITRONICS Queue Head
CMHEAD	COGNITRONICS Queue Head
CMSWORD	COGNITRONICS Output Address
FCONT3	Flight Control Word for T99DI3
GOFLAG	Exercise Start Flag
PAD	Roll Rate (degrees/second)
PHIS	Bank Angle (degrees)
PMSG 17	KBP Message INPUT STUDENT FILE DATA

PMSG T3	KBP Message CRASH
PMSG T4	KBP Message DEPRESS THE RESET-TO-ZERO CONSOLE BUTTON
PSI	F-4 Heading (degrees)
PSIAFT	Desired AFT Heading
RCIS	Average Rate-of-Climb
SPSI	Average Heading
TERMCODE	Exercise Termination Code

2. External Outputs: None

g. Program Entrances

1. BAL, 15 ATE$F — ATE/AFT Foreground Executive
2. BAL, 15 ATE$B — ATE/AFT Background Executive
3. BAL, 15 RESET$1 — RESET-TO-ZERO Routine
4. BAL, 15 CRASH$1 — Reset Crash Conditions Routine
5. ACT P1, P2, ... P_N — Meta-symbol procedure to activate an ATE/AFT program
6. DACT P1, P2, ... P_N — Meta-symbol procedure to deactivate an ATE/AFT program

h. Exits

1. B *FSAV15 — Foreground Executive (calling location +1)
2. B *15 — Background Executive (calling location +1)
3. B *RESEX — RESET-TO-ZERO Exit (calling location +1)
4. B *CRAEX — Reset Crash Conditions Exit (calling location +1)

5. B *15 ACT Exit (calling location +N)**

6. B *15 DACT Exit (calling location +N)**

$$** \ N = 1 + \left(\frac{\text{Number of Programs}}{4} \right) \text{(truncated)}$$

i. Subroutines Called

1. CLRATE Used by RESET$1 and CRASH$1 to clear all ATE/AFT programs (I/O routines excepted) in the Program Status Tables.

2. POUT:1 Keyboard/Printer Output

3. QPURGE Purge COGNITRONICS Queue

4. ARCSIN Computes heading and bank angles

j. Memory Requirements

1. Instruction 266

2. Data 39

k. Type of Program Module

1.	ATE$F	ATE Foreground
2.	ATE$B	ATE Background
3.	RESET$1	Subroutine callable from F4 MCC module
4.	CRASH$1	Subroutine callable from F4 MCC module
5.	ACT (CHGSTA:A)	Meta-symbol procedure callable from ATE/AFT Foreground or Background
6.	DACT (CHGSTA:D)	Meta-symbol Procedure callable from ATE/AFT Foreground or Background

l. Flow Charts. See figure 2-6.

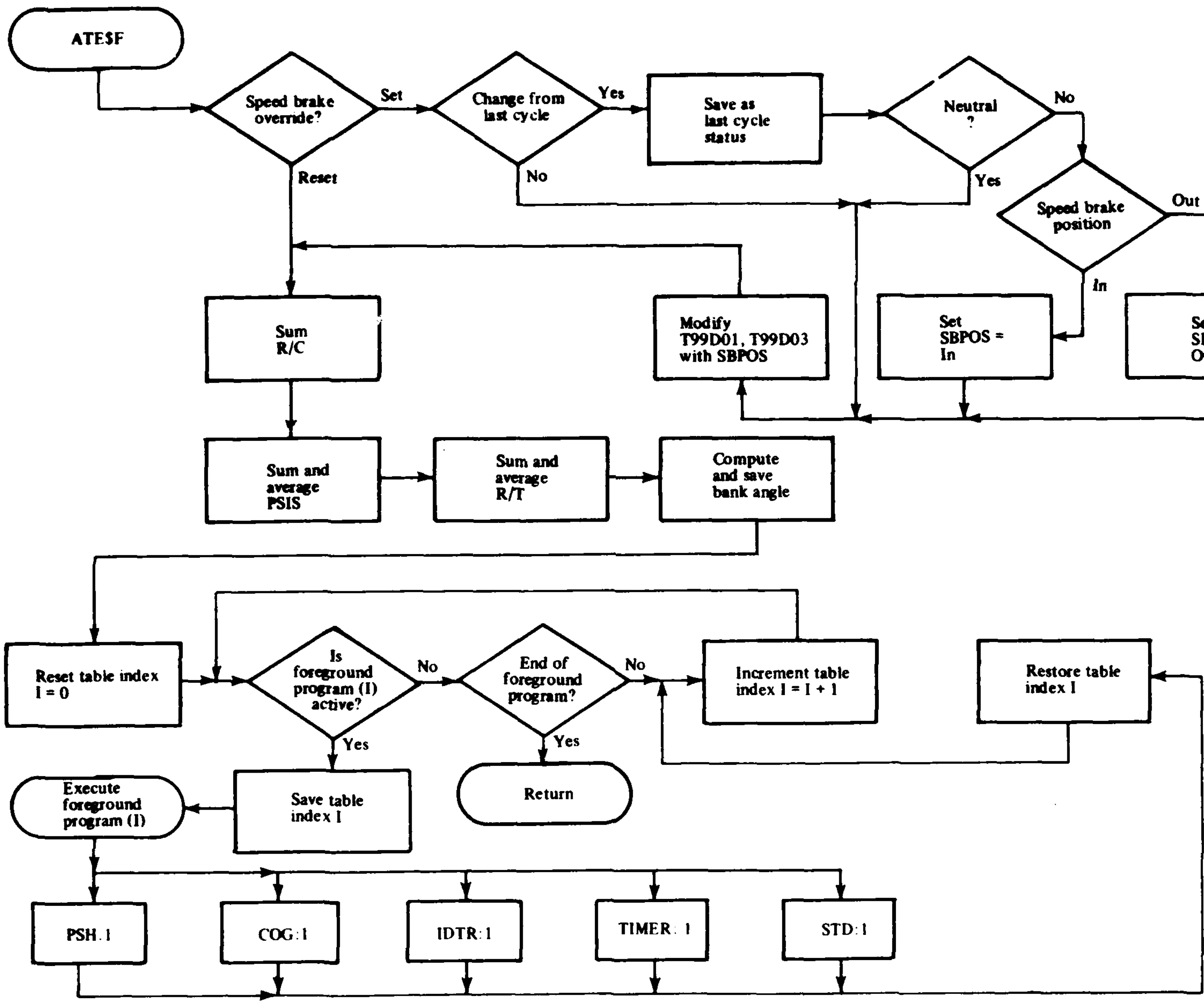


Figure 2-6. ATE/AFT Executive Routines (Sheet 1 of 4)

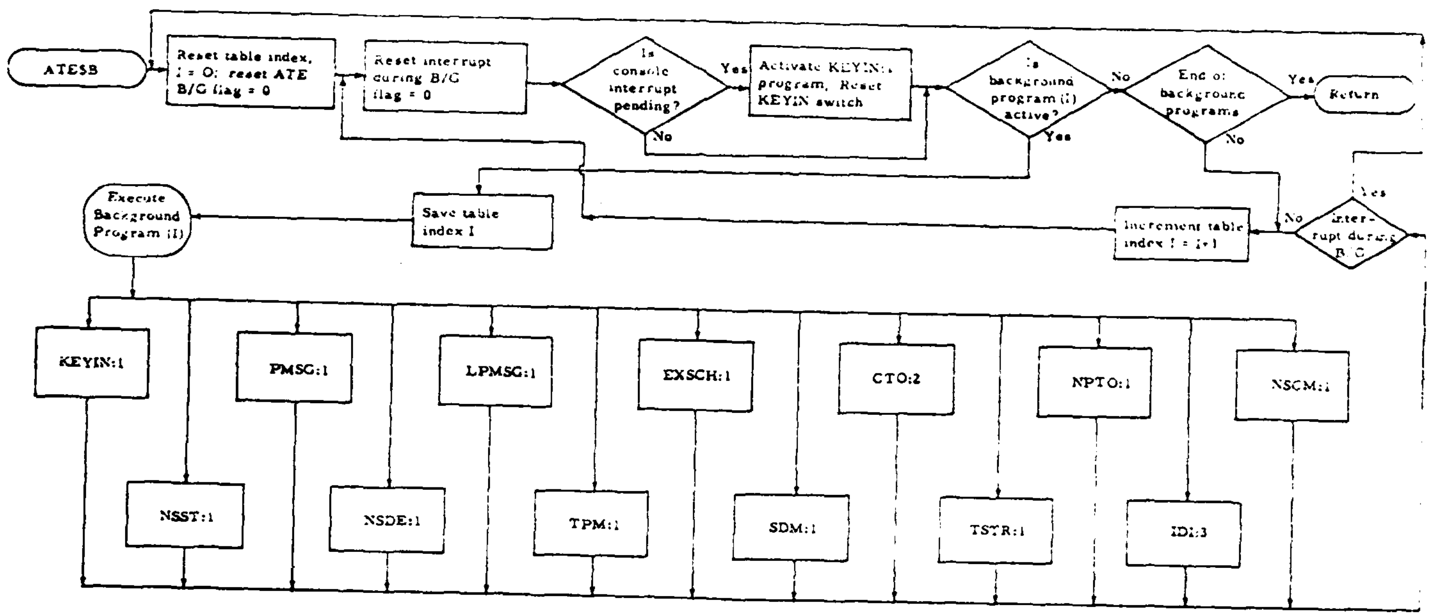


Figure 2-6. ATE/AFT Executive Routines (Sheet 2 of 4)

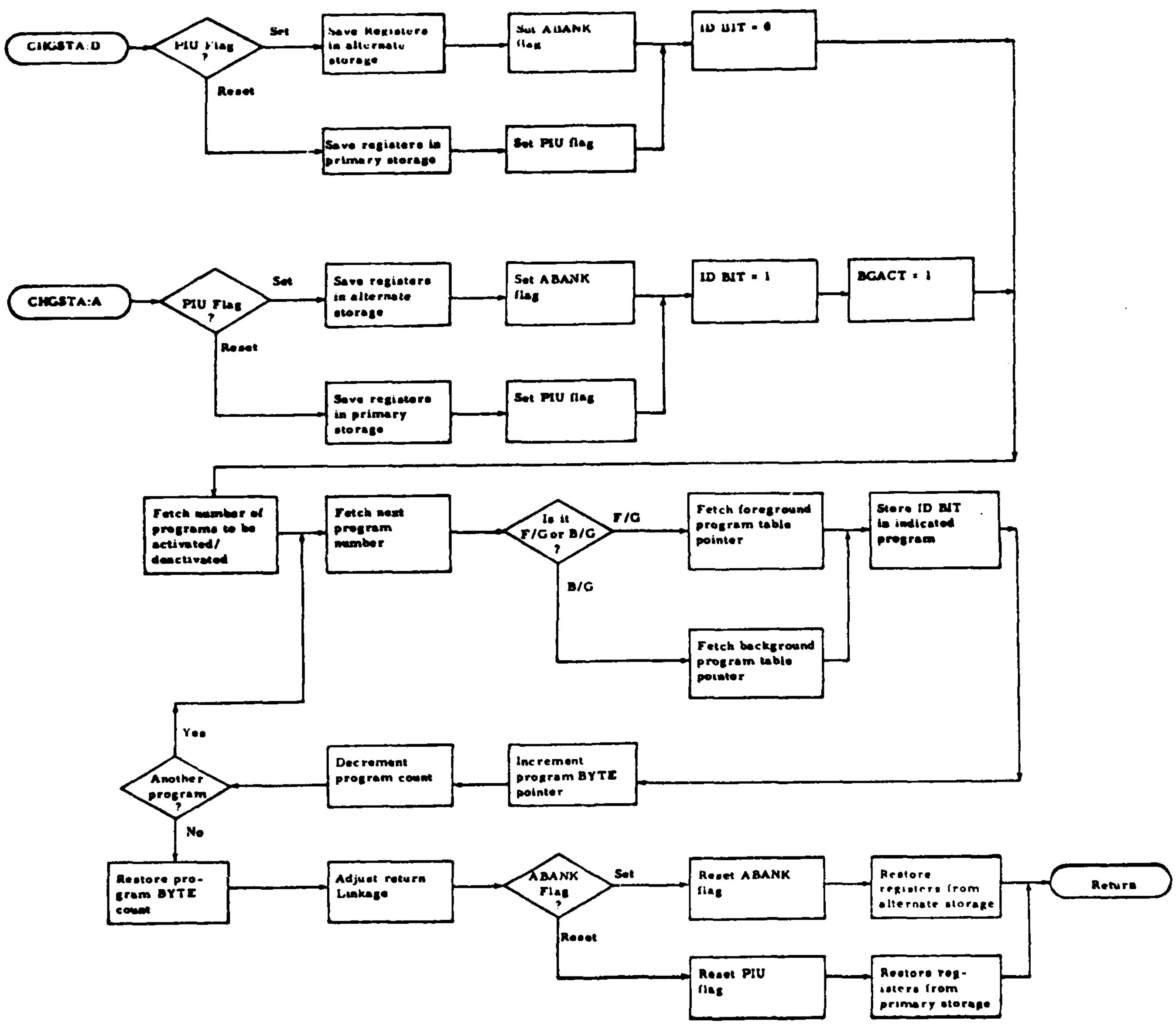


Figure 2-6. ATE/AFT Executive Routines (Sheet 3 of 4)

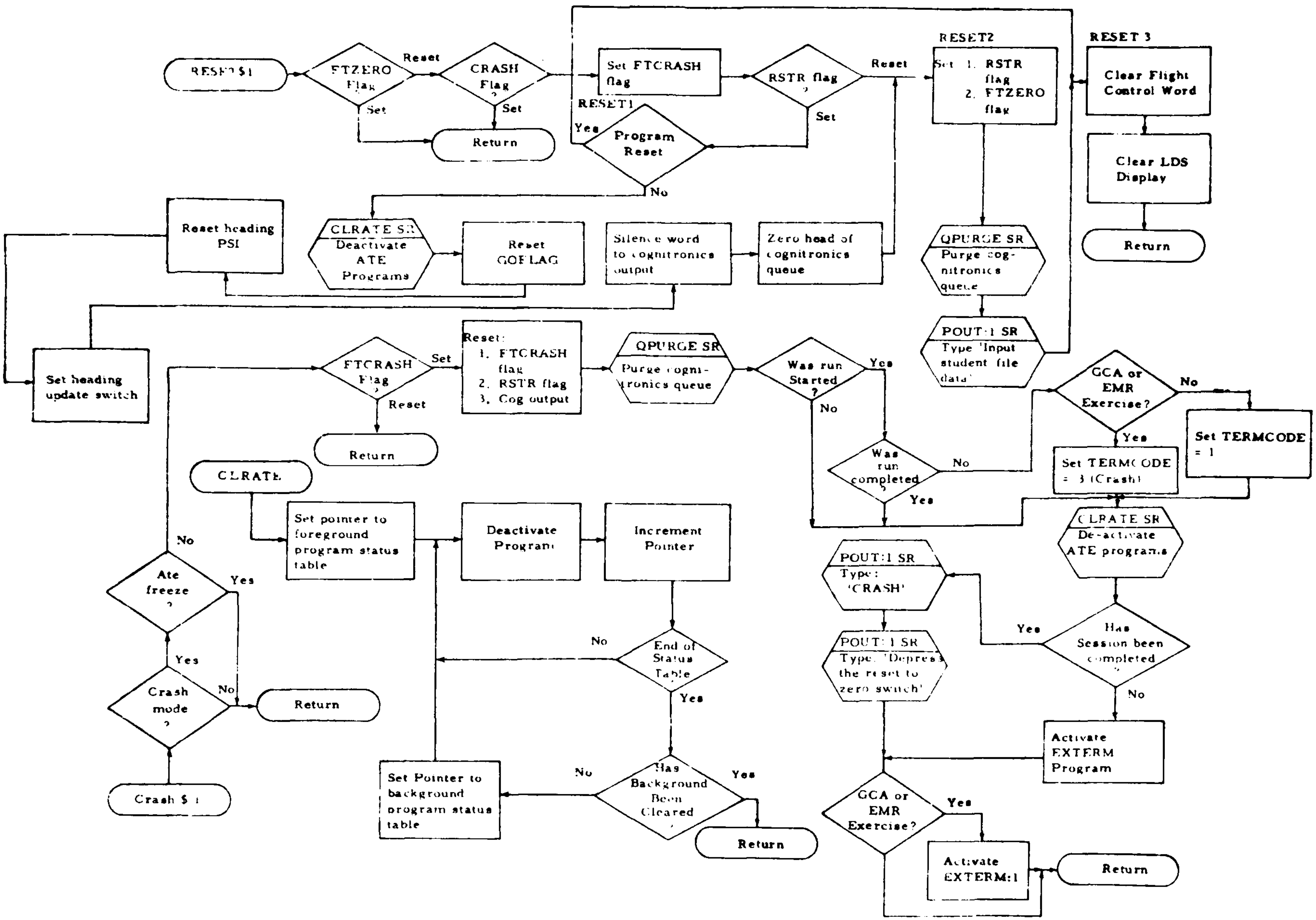


Figure 2-6. ATE/AFT Executive Routines (Sheet 4 of 4)

2.8 COGNITRONICS MESSAGE PROCESSOR

a. Program Module Name. COGNITRONICS Message Processor (COG)

b. Purpose. The purpose of COG is to monitor and control all COGNITRONICS output messages.

c. Requirements. The COG program module is required:

1. To select and place in the COGNITRONICS output buffer the next COGNITRONICS output word address.

2. To deactivate the COGNITRONICS output routines if no further message words are awaiting output.

3. To manipulate the COGNITRONICS message queue to ensure the correct order and priority of message output.

4. To insert new messages into the COGNITRONICS queue in order of priority, when required.

5. To purge the COGNITRONICS queue upon request.

d. Description. A description of the operation of the COGNITRONICS Speechmaker can be found in the COGNITRONICS Operation and Maintenance Manual. Tables 2-6 and 2-7 list the COGNITRONICS vocabulary addresses alphabetically and numerically respectively. Messages for the GCA and Emergency Procedures Exercises are assembled by the Meta-Symbol Procedure (PROC) 'CMSG' in this program. Figure 2-7 depicts the COGNITRONICS Output Message format generated by PROC 'CMSG.' The first word of each CMSG contains the priority, group, sequence number, and pointer to the next message in the output queue, if any. The first byte of the second word indicates the number of words contained in the message. All successive bytes contain the COGNITRONICS addresses of the message string. The 'CMSG' PROC is described further in paragraph 2.1.d.1.

Figure 2-8 is an example of the priority queuing for COGNITRONICS messages awaiting output. Messages awaiting output are queued together in order of priority by the QINSERT subroutine which is callable from any ATE Background program module. Once a message has started being processed for output by the COG:1 routine it cannot be replaced at the top of the message queue by another message which is to be inserted in the queue. The COG:1 routine interrogates the KCOG counter (set by the

Table 2-6. COGNITRONICS Vocabulary (Alphabetic)

21 above	A5 clear	6B feet	70 is
61 acknowledge	66 clearance	0E fifty	B0 knots
A1 add	26 cleared	0C fifteen	31 land
24 adjust	A6 climb	AB final	32 leg
13 ahead	27 complete	07 five	B1 left
4A air	A7 contact	56 flaps	18 level
62 altitude	5E control	2C flight	72 line
A2 and	54 correct	30 formation	0B list
4B angels	A8 correction	86 four	B2 low
53 angle	29 course	6C from	73 mach
23 approach	22 craft	AC further	33 maintain
63 approaching	68 cross	2D glide	34 miles
93 assigned	94 decrease	4E go	B4 minimum
A3 at	A9 degrees	4C going	58 minus
6A attack	15 descend	6D good	35 minute
74 attitude	2A descent	49 half	75 missed
14 back	55 dive	78 hard	8E more
96 bank	AA down	2E heading	B5 nautical
A4 begin	95 ease	0A high	36 navy
64 behind	0D easy	AE hold	48 nine
25 below	4D ___ed	8C holding	76 no
8A brake	08 eight	2F hundred	19 nose
88 ___board	8B engine	6F if	37 not
11 bomb	8D ___er	AF in	B6 now
3C ___bound	16 establish	97 increase	B7 of
28 center	2B execute	57 ___ing	59 off
65 check	1F fast	44 I. P.	77 okay

Table 2-6. COGNITRONICS Vocabulary (Alphabetic) (Continued)

38 on	9A set	01 time	05 your
85 one	87 seven	6E (tone 1000 Hz)	45 zero
50 out	BD sight	A0 touchdown	92 zulu
B8 over	00 (silence)	41 transmission	
39 path	40 (silence)	60 trim	
79 pattern	80 (silence)	81 turn	
B9 per	47 six	5C twenty	
0F percent	3E slightly	06 two	
3A pitch	9E slow	71 ___ty	
9F place	5B speed	02 under	
5F plan	17 standard	1D up	
1A plus	7E star	51 vary	
7A point	9B start	42 vector	
4F port	7D steady	82 visual	
5A power	91 steer	03 visually	
BA precision	1C stick	98 ___ward	
99 pressure	1B stop	5D watch	
89 radar	BE take	AD way	
3B rate	67 target	43 well	
7B received	69 ___teen	83 wheels	
10 reduce	90 ten	04 wind	
8F reverse	3F the	9D wings	
BB right	7F thirty	1E with	
7C roll	BF thousand	B3 wrong	
09 route	46 three	12 xray	
BC run	20 threshold	52 yankee	
3D seconds	9C throttles	84 yaw	

Table 2-7. COGNITRONICS Vocabulary (Hexadecimal)

00 (silence)	1E with	3C ___bound	5A power
01 time	1F fast	3D seconds	5B speed
02 under	20 threshold	3E slightly	5C twenty
03 visually	21 above	3F the	5D watch
04 wind	22 craft	40 (silence)	5E control
05 your	23 approach	41 transmission	5F plan
06 two	24 adjust	42 vector	60 trim
07 five	25 below	43 well	61 acknowledge
08 eight	26 cleared	44 I. P.	62 altitude
09 route	27 complete	45 zero	63 approaching
0A high	28 center	46 three	64 behind
0B list	29 course	47 six	65 check
0C fifteen	2A descent	48 nine	66 clearance
0D easy	2B execute	49 half	67 target
0E fifty	2C flight	4A air	68 cross
0F percent	2D glide	4B angels	69 ___teen
10 reduce	2E heading	4C going	6A attack
11 bomb	2F hundred	4D ___ed	6B feet
12 xray	30 formation	4E go	6C from
13 ahead	31 land	4F port	6D good
14 back	32 leg	50 out	6E (tone-1000 Hz)
15 descend	33 maintain	51 vary	6F if
16 establish	34 miles	52 yankee	70 is
17 standard	35 minute	53 angle	71 ___ty
18 level	36 navy	54 correct	72 line
19 nose	37 not	55 dive	73 mach
1A plus	38 on	56 flaps	74 attitude
1B stop	39 path	57 ___ing	75 missed
1C stick	3A pitch	58 minus	76 no
1D up	3B rate	59 off	77 okay

Table 2-7 COGNITRONICS Vocabulary (Hexadecimal) (Continued)

78 hard	96 bank	B4 minimum
79 pattern	97 increase	B5 nautical
7A point	98 ___ward	B6 now
7B received	99 pressure	B7 of
7C roll	9A set	B8 over
7D steady	9B start	B9 per
7E star	9C throttles	BA precision
7F thirty	9D wings	BB right
80 (silence)	9E slow	BC run
81 turn	9F place	BD sight
82 visual	A0 touchdown	BE take
83 wheels	A1 add	BF thousand
84 yaw	A2 and	
85 one	A3 at	
86 four	A4 begin	
87 seven	A5 clear	
88 ___board	A6 climb	
89 radar	A7 contact	
8A brake	A8 correction	
8B engine	A9 degrees	
8C holding	AA down	
8D ___er	AB final	
8E more	AC further	
8F reverse	AD way	
90 ten	AE hold	
91 steer	AF in	
92 zulu	B0 knots	
93 assigned	B1 left	
94 decrease	B2 low	
95 ease	B3 wrong	

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31

P	GG	SN	PTR	
BC		CWA(1)	CWA(2)	CWA (3)
CWA (4)		CWA(5)	CWA(6)	CWA (7)
⋮				
CWA(BC-1)		CWA(BC)	00000000	00000000

P Is the COGNITRONICS Message Priority (1 through 10). Lowest Numbers Indicate Highest Output Priority.
GG Is the Message Group
SN Is the Message Sequence Number
PTR Is the Pointer to the next Highest Priority in the Output Message Queue. If PTR 0, Either Message is at Tail of Queue or Message is not Currently in Queue.
BC Is the Byte Count of the COGNITRONICS Word Addresses (CWA) which Compose the Message.
CWA Is the COGNITRONICS Word Address (See Tables 2-6 and 2-7)

Figure 2-7. COGNITRONICS Output Message Format

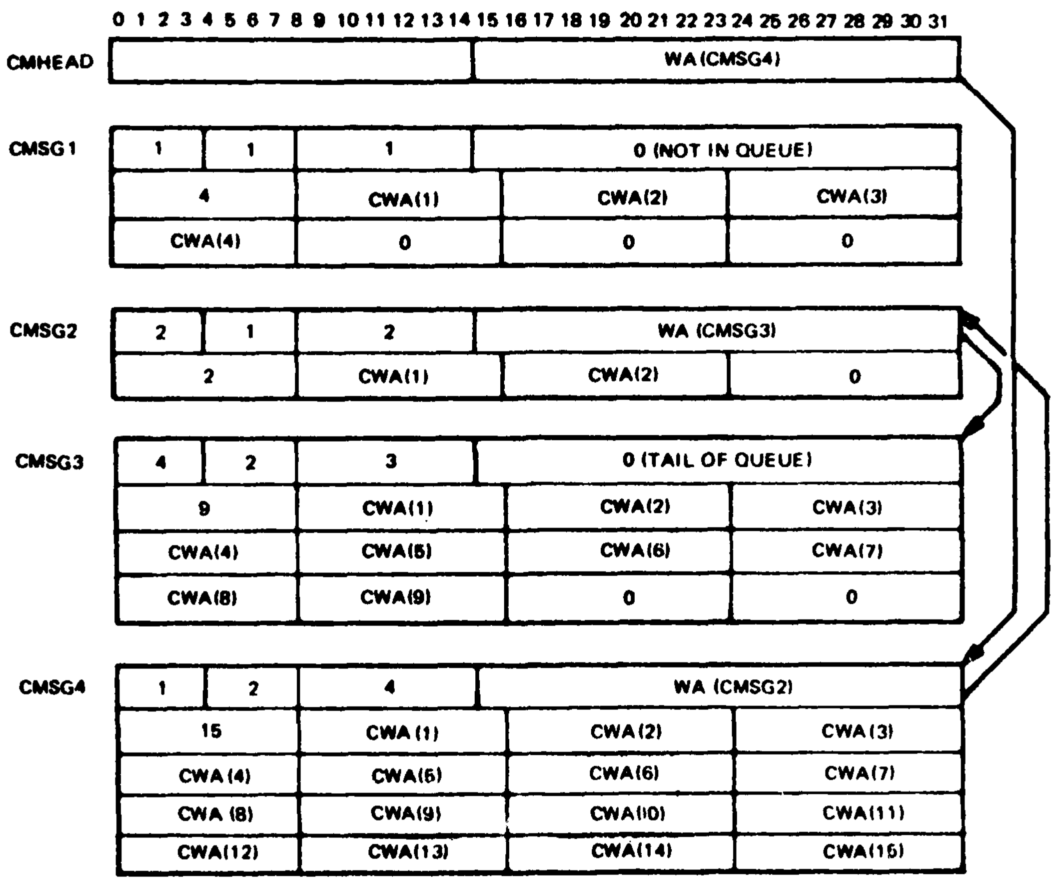


Figure 2-8. Example of COGNITRONICS Message Queue

COGNITRONICS interrupt in the MCC module) to determine if the COGNITRONICS is ready to accept the next word address in the message queue. COG:1 is activated whenever a call is made to the QINSERT routine and is deactivated when the message queue is empty.

The QPURGE routine purges the COGNITRONICS message queue. It is usually employed whenever the run is terminated or when a CRASH or RESET-TO-ZERO condition is detected.

e. Inputs

1. Internal Inputs:

CMHADDR	Address of COGNITRONICS Queue Head
KCOG	COGNITRONICS Channel Counter

2. External Inputs: None

3. Constants: (Refer to Table 2-8 for list of the Nav/Strike COGNITRONICS messages.)

CMBCOUNT	CMSG Byte Count
CMBYTEX	CMSG Byte Index
CMFIRST	First Time COGNITRONICS Message Switch
CMHEAD	COGNITRONICS Message Queue Head

f. Outputs

1. Internal Outputs:

CMSWORD	COGNITRONICS Output Address
GRPSW	COGNITRONICS Message Group Switches

2. External Outputs: None

g. Program Entrances

1.	BAL, 15 COG:1	COGNITRONICS Word Output Routine
2.	BAL, 15 QINSERT DATA WA (CMSG)	Insert Message in COGNITRONICS Message Queue
3.	BAL, 15 QPURGE	Purge COGNITRONICS Queue

Table 2-8. Navigation/Strike COGNITRONICS Messages

Symbol	Message
ABMSG	VECTOR (HEADING) FOR ATTACK HEADING; SPEED POINT NINE, ANGELS TWENTY; TARGET ONE AIRCRAFT, ALTITUDE TWENTY THOUSAND; TARGET HEADING 360, SPEED POINT NINE.
ABMSGR	ATTACK HEADING IS (HEADING), ANGELS TWENTY, SPEED POINT NINE.
ADMSG	STEADY ON ATTACK VECTOR, TARGET (BEARING) AT (RANGE), TAKE RADAR CONTROL FOR FINAL ATTACK.
BKMSG01	PORT.
BKMSG02	STARBOARD.
BKMSG07	HARD PORT.
BKMSG08	HARD STARBOARD.
BKMSG13	HARD PORT SIXTY.
BKMSG14	HARD STARBOARD SIXTY.
CAL13	ASSIGNED ALTITUDE IS TWENTY THOUSAND FEET.
CALAT	CHECK ALTITUDE.
CBMSG01	CLIMB TO TWENTY THOUSAND FEET ON HEADING 045 DEGREES AT 100 PERCENT POWER AND 350 KNOTS.
CBMSG07A	CLIMB TO 15 HUNDRED FEET, HEADING 045 DEGREES, SPEED 280 KNOTS.
CBMSG07B	REVERSE COURSE ONE EIGHTY, CLIMB TO TWENTY THOUSAND FEET AT 100 PERCENT POWER AND 350 KNOTS.
CBMSG13A	CLIMB TO 15 HUNDRED FEET, HEADING 360 DEGREES, SPEED 350 KNOTS.
CBMSG13B	VECTOR PORT 330, CLIMBING TO TWENTY THOUSAND FEET AT 100 PERCENT POWER AND MACH POINT NINE.
CBMSG13C	VECTOR STARBOARD 060.
CMC07D	ASSIGNED MACH IS POINT EIGHT.
CMC13	ASSIGNED MACH IS POINT NINE.
CMCAT	CHECK MACH.
CMSGAC	ATTACK COMPLETED, BREAK (PORT/STARBOARD).
CMSGBRK	BREAK OFF ATTACK TO (PORT/STARBOARD).

Table 2-8. Navigation/Strike COGNITRONICS Messages (Continued)

Symbol	Message
CMSGCC	CLIMB COMPLETED.
CMSGDC	DESCENT COMPLETED.
CMSGEBK	EASE BANK.
CMSGHBK	HOLD BANK.
CMSGIBK	HARDER BANK.
CMSGT03	ADJUSTING AIRCRAFT FOR FLIGHT.
CPS01	ASSIGNED HEADING IS 045 DEGREES.
CVG01	ASSIGNED AIRSPEED IS 350 KNOTS.
CVG01D	ASSIGNED AIRSPEED IS 250 KNOTS.
CVG07C	ASSIGNED AIRSPEED IS 280 KNOTS.
CVG13D	ASSIGNED AIRSPEED IS 150 KNOTS.
DAL13M	ALTITUDE TOO LOW.
DAL13P	ALTITUDE TOO HIGH.
DBMSG01	POWER BACK, BEGIN DESCENT TO 15 HUNDRED FEET, MAINTAIN 250 KNOTS.
DBMSG07A	POWER 70 PERCENT, SPEED BRAKE OUT, BEGIN DESCENT MAINTAINING SPEED MACH POINT EIGHT.
DBMSG07B	REVERSE COURSE ONE EIGHTY, LEVEL OFF AT 15 HUNDRED FEET, SPEED 250 KNOTS.
DBMSG13A	STEER HEADING 330, MAINTAIN TWENTY THOUSAND, REDUCE POWER, HOLD SPEED 280 KNOTS.
DBMSG13B	TURN PORT HEADING 270 DEGREES, POWER 70 PERCENT, SPEED BRAKE OUT AND BEGIN DESCENT AT THREE HUNDRED KNOTS.
DBMSG13C	TURN PORT HEADING 045 DEGREES AT THIRTY DEGREES BANK, LEVEL OFF AT 15 HUNDRED FEET, MAINTAIN HEADING AND SLOW TO 150 KNOTS.
DMC13M	MACH TOO SLOW.
DMC13P	MACH TOO FAST.
DPS01M	HEADING OFF TO PORT.
DPS01P	HEADING OFF TO STARBOARD.

Table 2-8. Navigation/Strike COGNITRONICS Messages (Continued)

Symbol	Message
DVG01M	AIRSPEED TOO SLOW.
DVG01P	AIRSPEED TOO FAST.

h. <u>Exits</u>

1. B *COGEX — COG:1 Exit (calling location +1)
2. B *15 — QINSERT Exit (calling location +2)
3. B *15 — QPURGE Exit (calling location +1)

i. <u>Subroutines Called</u>: None

j. <u>Memory Requirements</u>

Instructions	143
Data	34

k. <u>Type of Program Module</u>

1. COG:1 — ATE Foreground Program
2. QINSERT — Subroutine callable from ATE Background Programs.
3. QPURGE — Subroutine callable from ATE Foreground or Background Programs.

l. <u>Flow Charts</u>. See Figures 2-9 and 2-10.

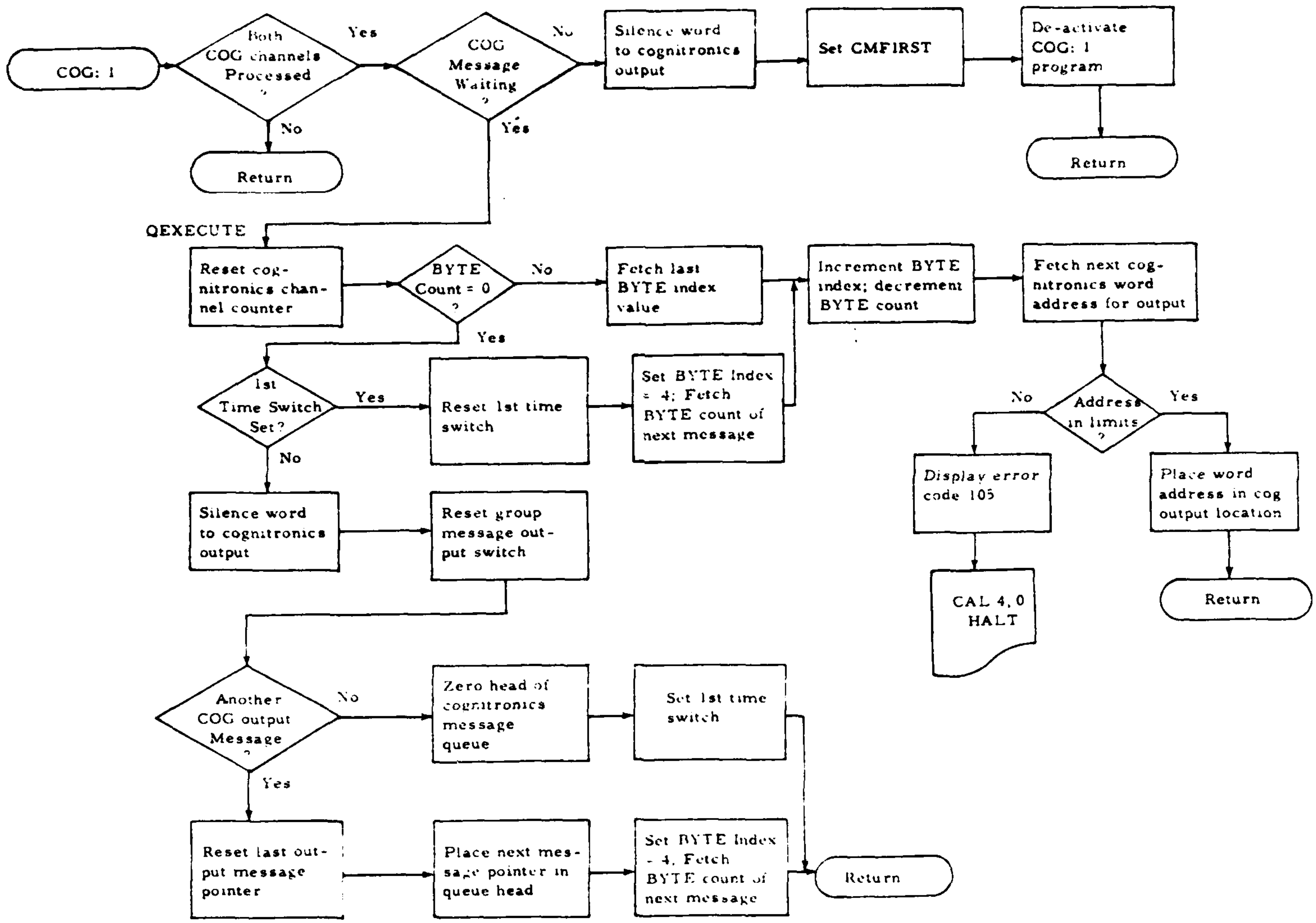


Figure 2-9. COGNITRONICS Message Processor

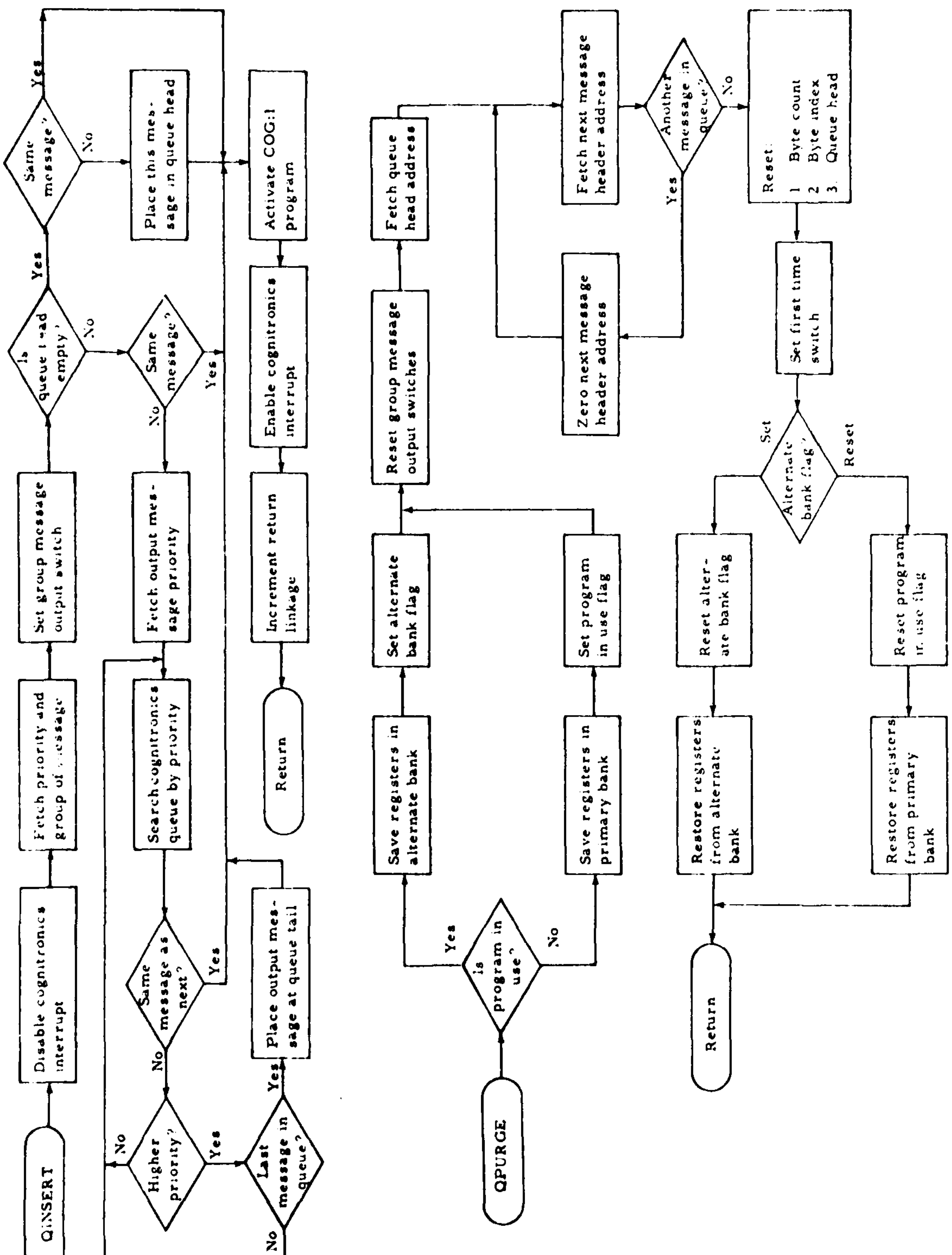


Figure 2-10. COGNITRONICS Message Queuing Subroutine

2.9 IDIIOM DISPLAY LIST TRANSMISSION

a. Program Module Name. IDIIOM Display List Transmission (IDTR)

b. Purpose. The purpose of IDTR is to output to the IDIIOM the next set of display list parameters.

c. Requirements. IDTR is required to:

1. Test the IDIIOM to ensure it is ready to receive the next set of data.

2. If ready, to output the next set of display list parameters.

d. Description. IDTR is a short routine that initiates the I/O to the IDIIOM for the next set of display list parameters. WRIDI is the double word address which specifies the output buffer and number of bytes of data to be transferred. WRIDI and the contents of the output buffer are determined in the IDI Background Program module.

e. Inputs

1. Internal Inputs:

   WRIDI — IDIIOM Output Doubleword Command

2. External Inputs: None

3. Constants: None

f. Outputs

1. Internal Outputs: None

2. External Outputs: None

g. Program Entrances

   BAL. 15 — IDTR: 1

h. Exits

   B *15 — (calling location + 1)

i. Subroutines Used. None

j. Memory Requirements

1. Instructions 7

2. Data 1

k. Type of Program Module. Foreground.

l. Flow Charts. See Figure 2-11.

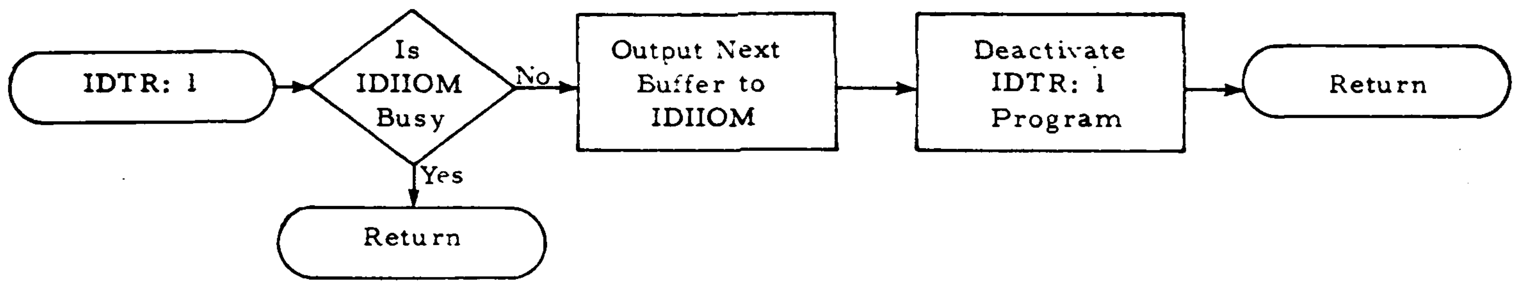


Figure 2-11. IDIIOM Display List Transmission

2.10 PSEUDO-HEADING ROUTINE

a. Program Module Name. Pseudo-Heading Computation (PSH)

b. Purpose. The purpose of the PSH module is to provide a rapid and computer-controlled means of initializing the assigned heading prior to the next run.

c. Requirements. The PSH module is required to:

1. Compute a pseudo-heading for use by the ATE/AFT and instrument output programs which is a function of the F-4 flight program but can be altered without affecting the F-4 flight equations.

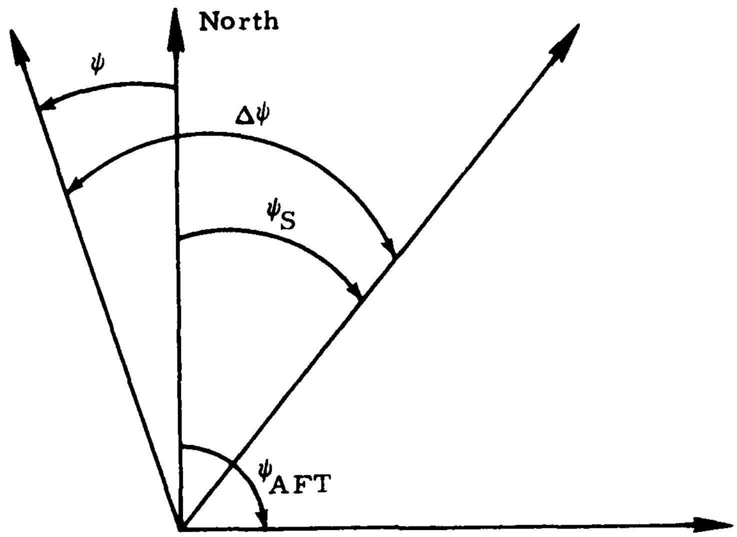
2. Alter the pseudo-heading to specified values, upon request.

d. Description. A pseudo-heading was incorporated into the AFT program to permit the instrument heading to be expeditiously initialized prior to the start of the next training task. This technique eliminates the requirement that the trainee fly the aircraft to the assigned heading. Figure 2-12 illustrates the pseudo-heading geometry. The actual F-4 heading is the angle ψ while the pseudo-heading is angle ψ_S. During a run, the pseudo-heading is updated by keeping the angle $\Delta\psi$ constant as ψ changes. Prior to each new run, ψ_S is driven to the new assigned heading ψ_{AFT} and the new $\Delta\psi$ value is recorded for use throughout the new run. In order to have the pseudo-heading indicated on the console and cockpit instruments, a portion of the PSH routine is devoted to modification of the F-4 Aircraft Instrument (ACI) program module. The PSH:Z subroutine is called in the F-4 initialization process to permit pseudo-heading outputs and a speed brake override. The speed brake is used by the trainee to control the start of the exercise (IFM maneuver) and the program alterations remove the aerodynamic effects that speed brake manipulation would normally have upon the F-4 equations.

e. Inputs

1. Internal Inputs:

PSIINC	PSI Increment
PSIS	Last Pseudo-Heading



ψ = F-4 Flight Equation Heading

ψ_{AFT} = Desired AFT Heading

ψ_S = Pseudo-Heading (This cycle)

Figure 2-12. Pseudo-Heading Geometry

2. External Inputs:

ACIINST1	ACI Module Alteration
ACIINST2	ACI Module Alteration
ACIINST3	ACI Module Alteration
ACIINST4	ACI Module Alteration
LCEINST1	LCE Module Alteration
PSI	Aircraft Heading
COSPSI	Cosine Heading Output
SINPSI	Sine Heading Output

3. Constants: None

f. Outputs

1. Internal Outputs:

DMACHVAL	Desired Mach Value
PSIAFT	Desired AFT Heading
PSIB	Base Heading
PSIS	Last Pseudo-Heading
PSIUP	Heading Update Switch

2. External Outputs:

COSPSIF4	Cosine of Heading
SINPSIF4	Sine of Heading
T99AO26	Analog Output Word 26
T99AO27	Analog Output Word 27

g. Program Entrances. BAL, 15 PSH:1

h. Exits. B *PSHRET (calling location +1)

i. Subroutines Called

1. COSD$1 Cosine

2. SIND$1 Sine

j. Memory Requirements

1. Instructions 99

2. Data 4

k. <u>Type of Program Module</u>. ATE Foreground

l. <u>Flow Charts.</u> See figure 2-13.

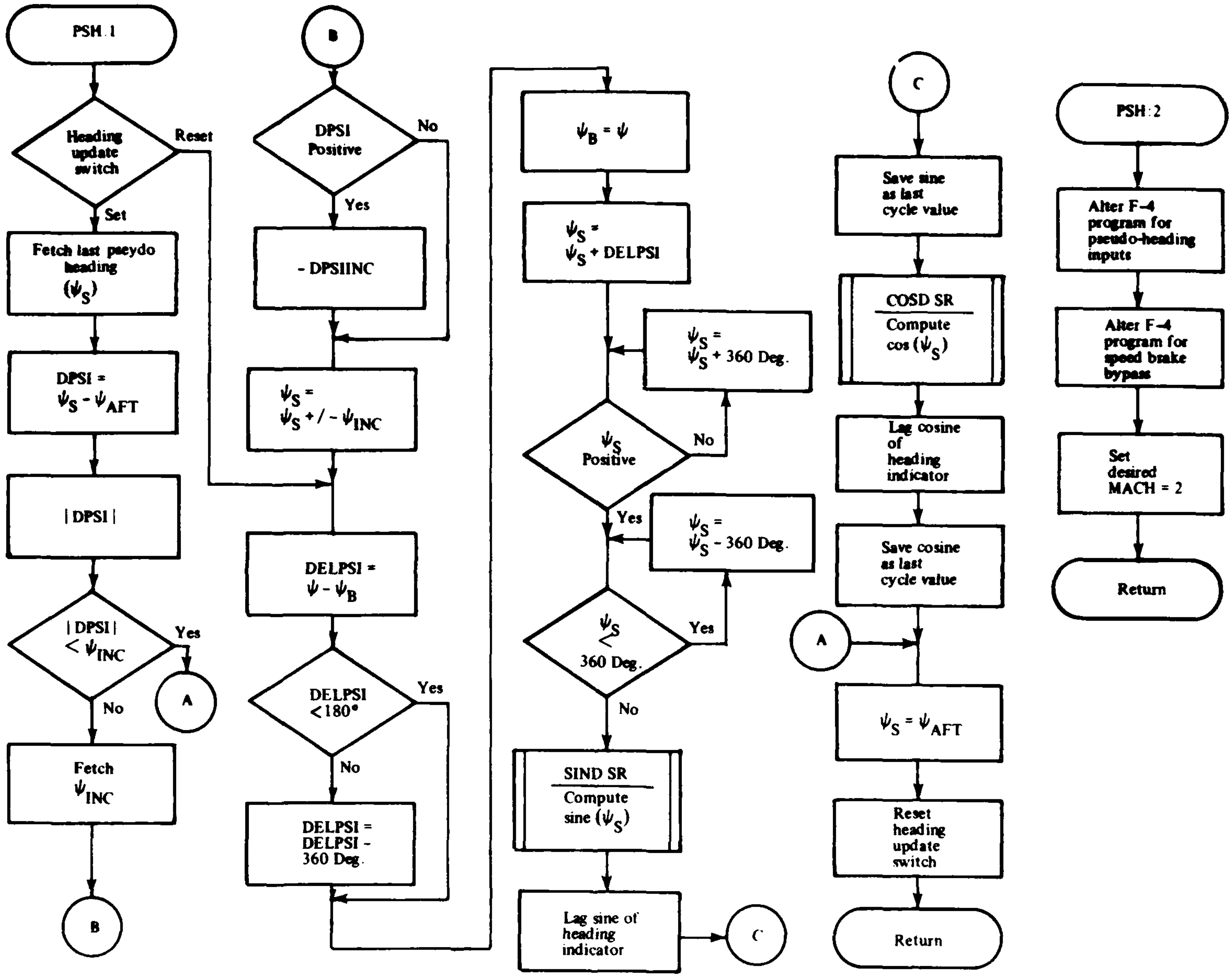


Figure 2-13. Pseudo-Heading Computation

2.11 STRIKE TARGET DYNAMICS

a. Program Module Name. Strike Target Dynamics (STD)

b. Purpose. The purpose of the STD module is to compute the X, Y, and Z coordinates of the interceptor relative to the target aircraft.

c. Requirements. The STD module is required to:

1. Compute the initial position of the interceptor relative to the target aircraft.

2. Each cycle (50 ms), update the X, Y, and Z coordinates of the interceptor relative to the target aircraft.

3. Compute the X and Y rate terms ($\dot{X}$, $\dot{Y}$).

4. Compute the horizontal and vertical look-angles with respect to the interceptor.

5. Set the final attack phase if the look-angles and relative position data are within prescribed limits.

6. Terminate the run if "lock-on" is lost.

d. Description. Once the strike task is initiated, the initial position of the interceptor from the target aircraft is computed. These computations are performed on the first pass through the STD module. The initial interceptor position (XI, YI, ZI) is computed as follows:

$$XI = -R \cos \beta$$

$$YI = R \sin \beta$$

$$ZI = ZT - HI$$

where:

R Target range

β = Target bearing

ZT = Target altitude

HI = Interceptor altitude

Since it is desired to intercept the target approximately 1 nautical mile astern, the coordinates with respect to this intercept point are also computed. These coordinates are computed as:

$$XIG = XI - XC$$

$$YIG = YI$$

where:

XIG = X distance from desired intercept point (gate)

YIG = Y distance from desired intercept point (gate)

XC = distance of intercept point behind target

Figure 2-14 illustrates the STD geometry. During each subsequent pass through the STD module, the interceptor position is updated with respect to the target gate position.

$$\dot{XI} = VXT - ((\cos\psi)(VG)(PKGPD4))$$

$$\dot{YI} = VYT - ((\sin\psi)(VG)(PKGPD4))$$

$$XIG = XI\text{-}1 + \dot{XI}$$

$$YIG = YI\text{-}1 + \dot{YI}$$

$$ZIG = ZT\text{-}\dot{H}I$$

where:

I indicates the current cycle values

I-1 indicates the previous cycle values

VXT = target velocity, X direction

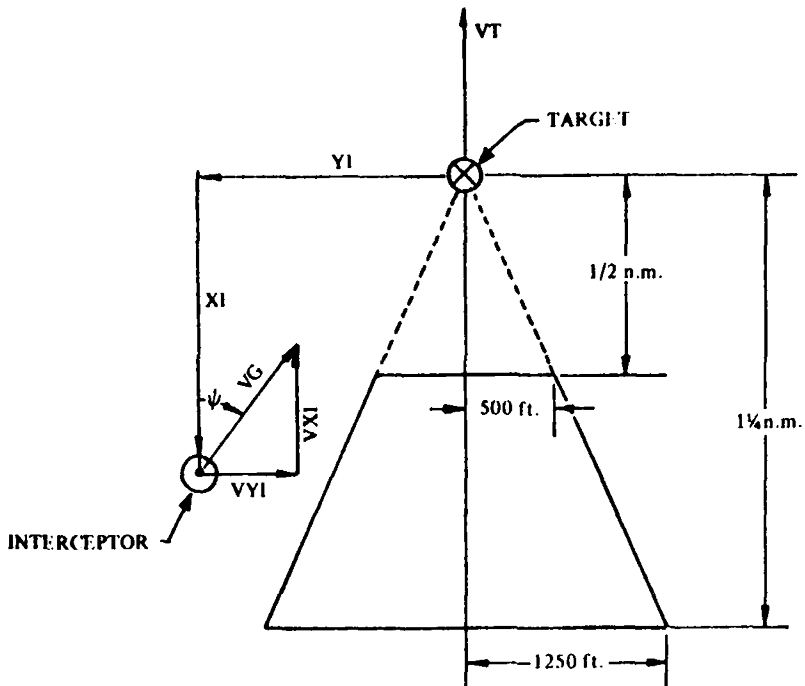
VYT = target velocity, Y direction

VG = interceptor velocity

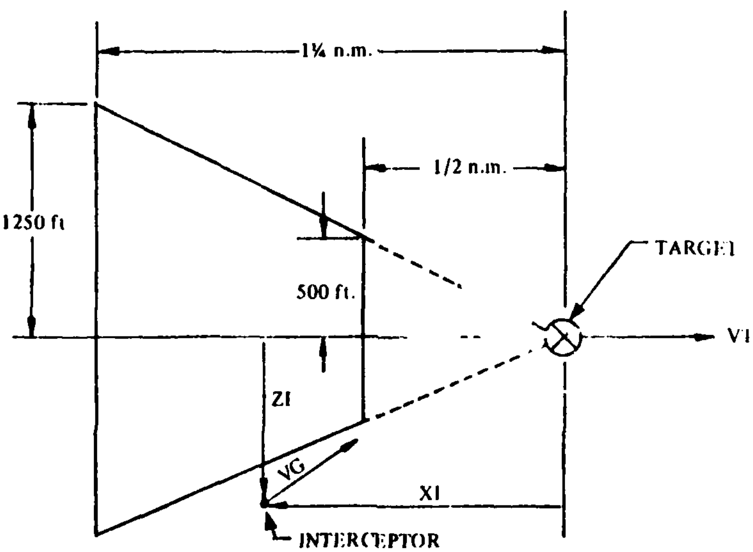
ψ = interceptor heading

PKGPD4 = convert knots to nautical miles

Once these computation are performed, the Check Look-Angle Subroutine (CHKLA) is called to see if the interceptor is within the target gate limits. If so, lock-on is established and the final attack (steering dot phase is entered) is initiated. The STD computations are continued, however, to ensure that lock-on is maintained.



X-Y PLANE (TOP VIEW)



X-Z PLANE (SIDE VIEW)

Figure 2-14. Strike Target Dynamics Geometry

The Check Look-Angle subroutine (CHKLA) computes the range and bearing of the target from the interceptor to see if lock-on is possible. The limits established require the range to be between 1/2 to 1-1/4 nautical miles, and the target to be within 15 degrees of the interceptor's nose.

e. <u>Inputs</u>

1. Internal Inputs:

COSLA	Cosine of Horizontal Look-Angle
COSPSI	Cosine of Heading
ITBRG	Initial Target Bearing
ITRNG	Initial Target Range
LFG	Chord Distance to Final Gate Position
PSIPD	Average Predicted Heading
PSIS	Pseudo-Heading
PSISU	Heading Update Switch
SINVA	Sine of Vertical Look-Angle
SINPSI	Sine of Heading
STDSW	1st Time STD switch
VXT	X Target Velocity
VYT	Y Target Velocity
XC	Gate Center Distance Behind Target
XGLIM	X Direction Gate Limits

2. External Inputs:

HI	Interceptor Altitude
VG	Interceptor Velocity

3. Constants:

D180	180
D360	360
KNMFT	Conversion factor, nautical miles to feet
PKGPD4	Conversion factor, knots to nautical miles

f. <u>Outputs</u>

1. Internal Outputs:

ATGATE	Attack Gate Penetration Switch
COSPD	Cosine of PSIPD
CSCSW	Compute Sin/Cos Switch
LA	Horizontal Look Angle (relative)
LOOKH	Horizontal Look Angle (true)
RUNTERM	Run Termination Switch
SINLOOK	Sine of LOOKH
SINPD	Sine of PSIPD
TSTK	Strike Run Time
XI	Distance Behind Target (this cycle)
XIG	Distance Behind Gate Position (this cycle)
XIGMI	Distance Behind Gate Position (last cycle)
ZI	Distance Below Target (this cycle)
ZT	Target Altitude

2. External Outputs: None

g. <u>Program Entrances.</u> BAL, 15 STD:1

h. Exits. B *STDRET

i. Subroutines Called.

ARCSIN	Arc Sine
COSD$1	Cosine
SIND$1	Sine

j. Memory Requirements.

Instructions	151
Data	7

k. Type of Program Module. Foreground

l. Flow Charts. See figures 2-15, 2-16.

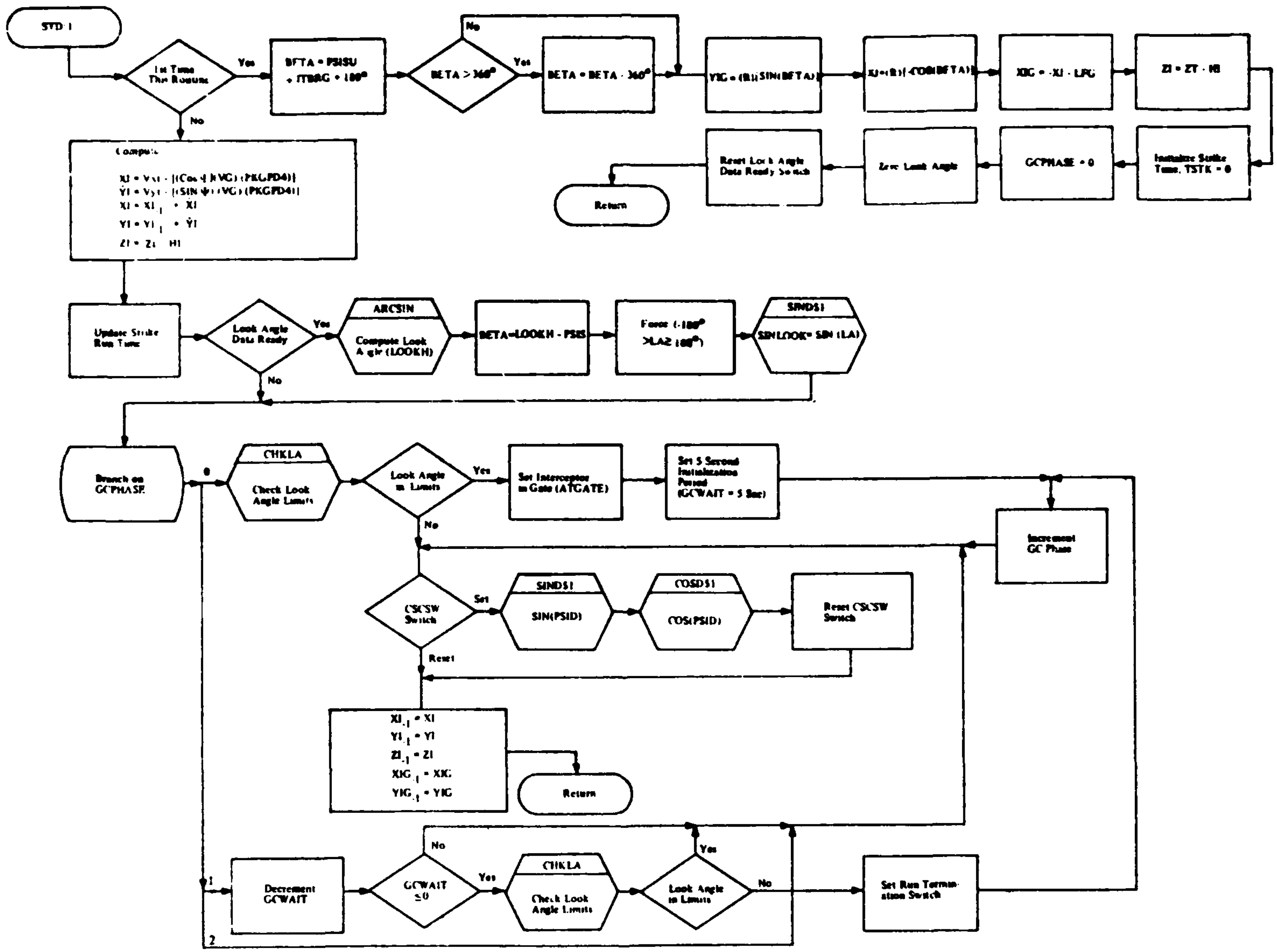


Figure 2-15. Strike Target Dynamics

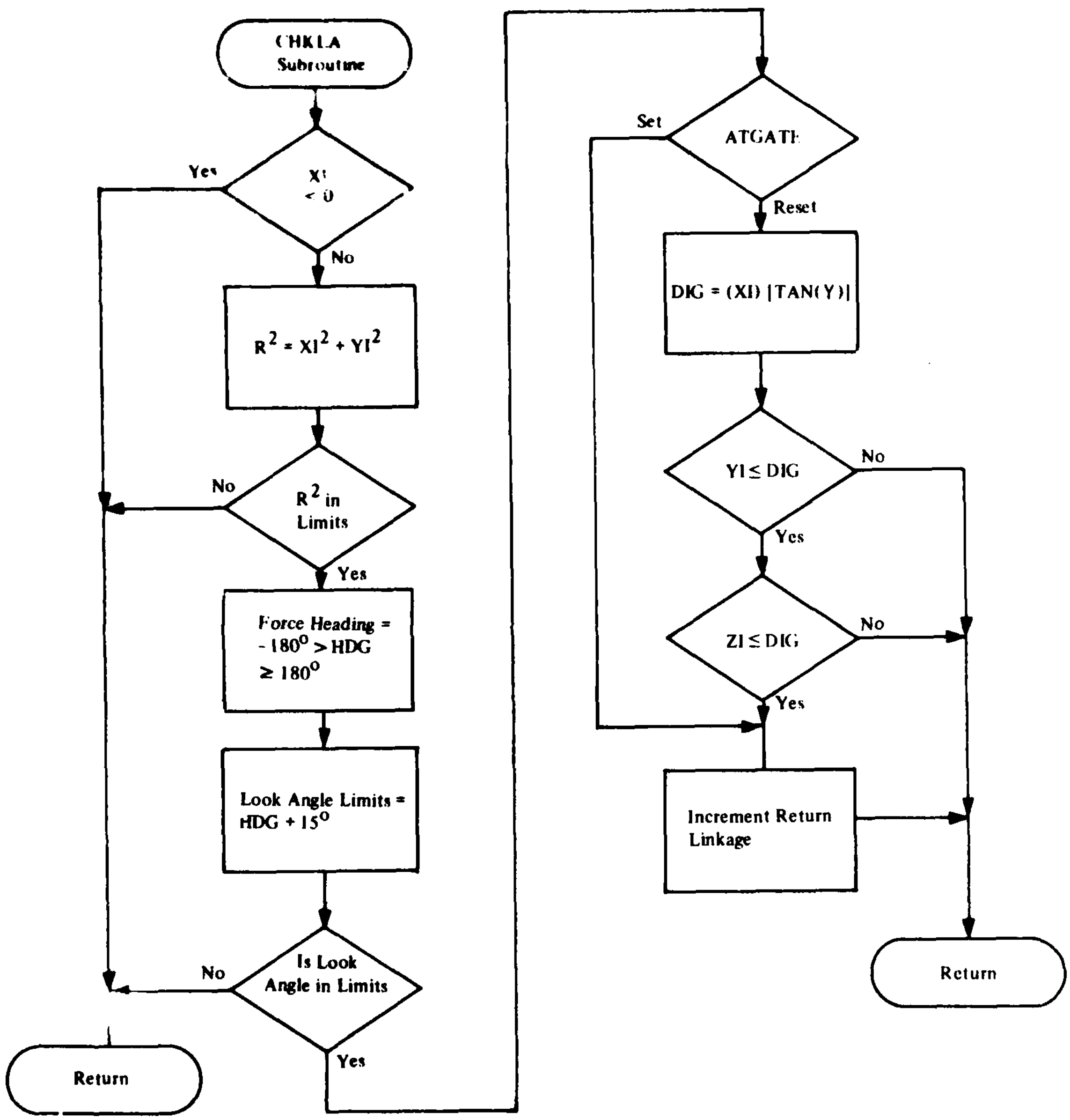


Figure 2-16. Look Angle/Gate Limit Check Subroutine

2-12 TIMING CONTROL

a. Program Module Name. Timing Control (TIMR)

b. Purpose. The purpose of the TIMR module is to provide real-time timers for ATE/AFT Background programs.

c. Requirements. The TIMR module is required to:

1. Decrement specified timers every program cycle (50 milliseconds)

2. Take prescribed actions when specified timers expire.

d. Description. The TIMR module is employed by certain ATE/AFT background routines to provide timing control for specified computations. At present, the two timers employed are a 1/2-second and a 1-second timer. The program is open-ended in that additional timers may be inserted as desired, providing they are multiples of the 50-millisecond program cycle. Timers are initialized when the TIMR module is actuated, and their count is started at this time. As each timer expires, the action directed by the appropriate Background Program is taken and the timer is reset. In the current ATE/AFT program, the 1/2-second timer is used to sample performance parameters and update the AFT display. The 1-second timer controls the ATE glide path parameter sample rate and the ATE display list update rate.

e. Inputs

1. Internal Inputs: None

2. External Inputs: CLOCK F4 Cycle Counter

3. Constants: TIM1ST 1st Time TIMR Entry Switch

f. Outputs

1. Internal Outputs:

IDITIME	IDIIOM Display List Update Switch
GPSTIME	Glide Path Para ıeters Sample Switch

GPMTIME General Performance Monitor Switch

PDISTIME Parameter Display Switch

2. External Outputs: None

g. Program Entrances. BAL, 15 TIMER:1

h. Exits. B *TIMEX (calling location +1)

i. Subroutines Called. None

j. Memory Requirements

1. Instructions 26

2. Data 10

k. Type of Program Module. Foreground Program

l. Flow Charts. See figure 2-17.

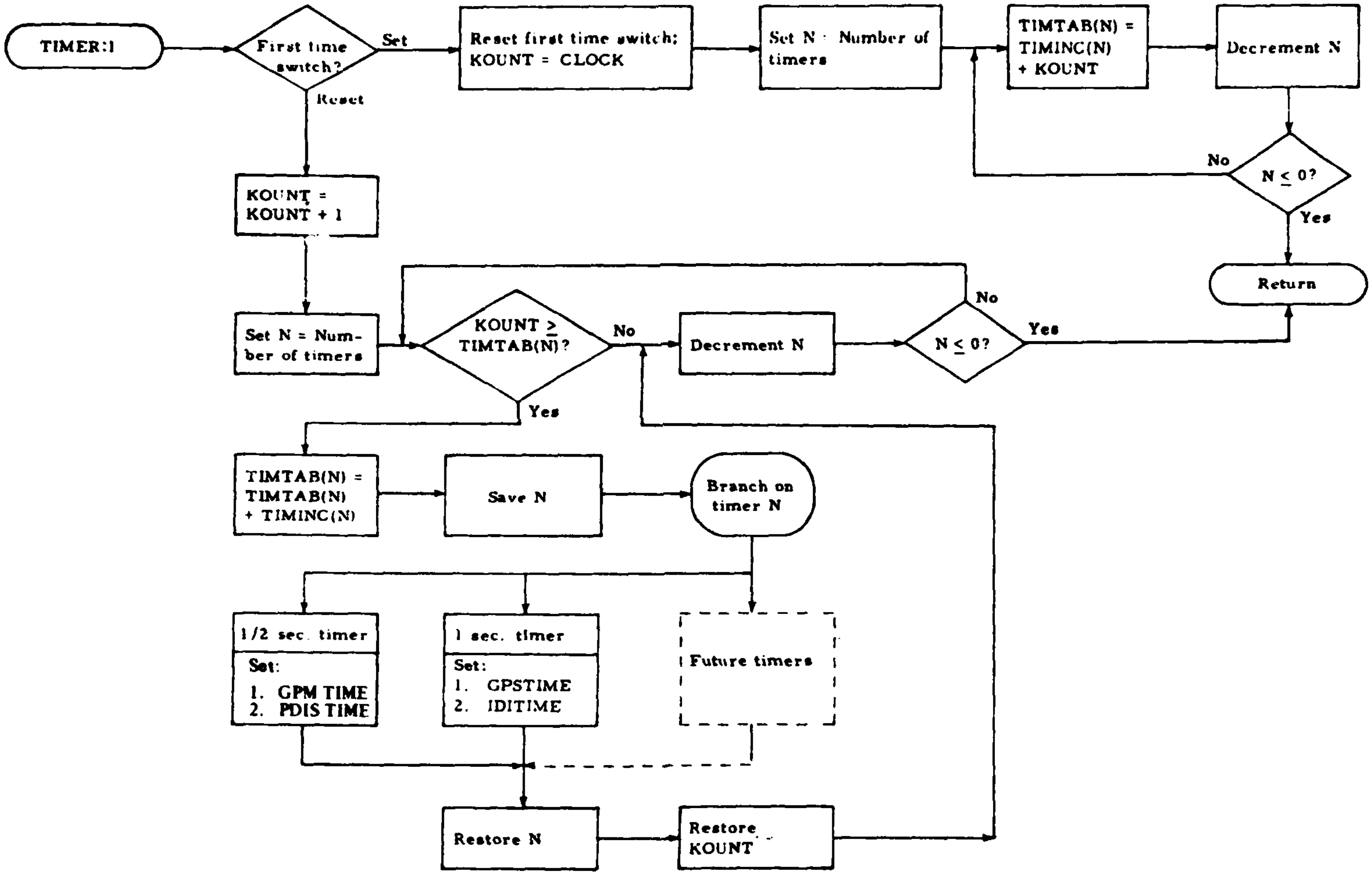


Figure 2-17. Timing Control

2.13 DATA INPUT/OUTPUT

a. Program Module Name. Data Input/Output Routines (I/O)

b. Purpose. The purpose of the I/O program module is to monitor and control all input/output between the ATE/AFT program and the SIGMA-7 peripherals (except for the IDIIOM and COGNITRONICS device which are handled by separate program modules).

c. Requirements. The I/O module must monitor and control:

1. The following keyboard inputs:

   (a) Student File Inputs

   (b) Exercise Parameter Changes

   (c) Absolute Program Patches

2. The ASR typewriter output

3. The Line Printer output.

d. Description. The I/O routine is composed of the following major routines:

1.	POUT:1	Insert typewriter output messages in the typewriter output message queue
2.	LPOUT:1	Insert line printer output messages in the line printer output message queue
3.	PMSG:1	Typewriter Output
4.	LPMSG:1	Line Printer Output
5.	KEYIN:1	Keyboard Inputs

The POUT:1 subroutine is called by ATE/AFT Program modules whenever a typewriter message is to be printed. POUT:1 queues all typewriter messages together in the format depicted in figure 2-18 on a first-in first-out basis. The I/O command doublewords for typewriter messages are identified by the alphabetic characters PMSG followed by a number. The message text buffer corresponding to the I/O command doubleword has the same symbol suffixed by the letter T; i.e., PMSG3T is the text associated with I/O command doubleword PMSG3. PMSGHEAD is a memory location reserved for the head of the message queue. It contains the address of the first I/O command doubleword to be output to the typewriter. If PMSGHEAD is zero, the message queue is empty. The first word of each text buffer is reserved for the pointer to the next I/O command doubleword in the message queue, if any. If this word is zero, it indicates either the tail of the queue or the message is not in the queue. (See figure 2-18.) ATE/AFT typewriter messages are listed in Table 2-9.

The PMSG:1 routine processes the typewriter message queue for output. It interrogates the Keyboard/Printer for a ready condition and, if ready, outputs the I/O command doubleword addressed by PMSGHEAD. It then removes the outgoing message from the queue and replaces it with the next message in the queuing list. When no more messages remain, PMSGHEAD is set to zero and the PMSG:1 routine is deactivated.

The LPOUT:1 and LPMSG:1 routines operate in a similar fashion to the POUT:1 and PMSG:1 routines except they operate on the line printer message queue. The line printer message queue has the same format as the typewriter message queue except I/O command doublewords and the message text buffers are preceded with the letter L; i.e., LPMSG3T is the line printer text associated with I/O command doubleword LPMSG3. Figure 2-19 is an example of the climb, attack, and descent tasks Line Printer output.

The KEYIN:1 routines process all keyboard inputs. Keyboard inputs for the ATE/AFT program are currently divided into the following three categories:

1. Student file inputs

2. Exercise parameter changes

3. Absolute program patches

NOTE: Before any keyboard input can be made, the lamp on the keyboard must be illuminated. This is accomplished by depressing the INTERRUPT button on the SIGMA-7 System Console.

All student file inputs are identified to the program by a $ (dollar sign). Up to 20 files can currently be maintained by the program.

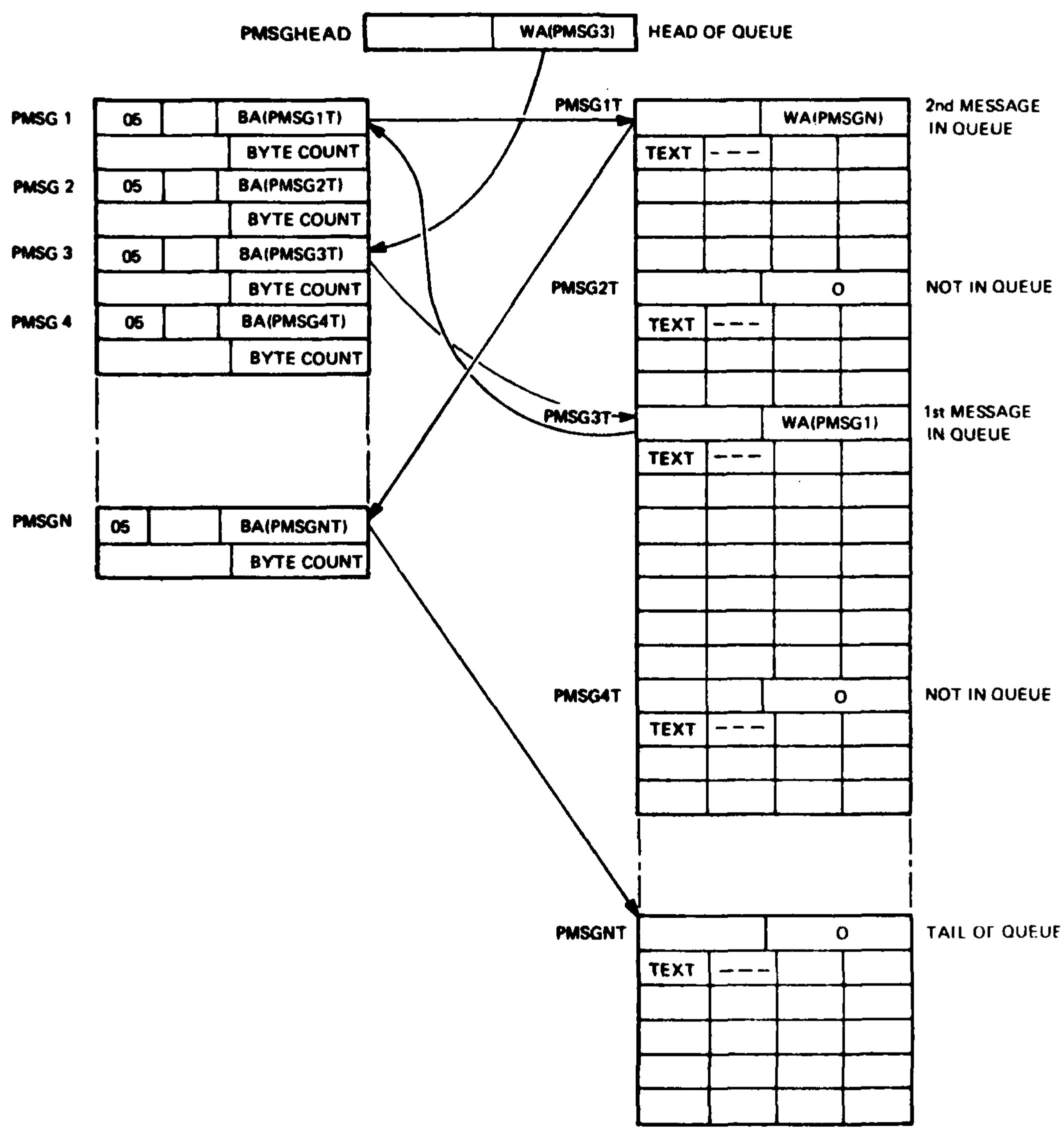


Figure 2-18. Example of Printer Message Queuing

Table 2-9. **Typewriter Message Formats**

PMSG2T	AUDIO BRIEFING COMPLETE
PMSG3T	CLEARED FOR TAKE-OFF
PMSG4T	IMPROPER TAKE-OFF CONFIGURATION
PMSG5T	AIRCRAFT AIRBORNE
PMSG6T	LANDING CHECK LIST NOT COMPLETE
PMSG9T	APPROACH COMPLETE
PMSG10T	CHECK WHEELS UP
PMSG11T	CHECK FLAP POSITION
PMSG12T	CHECK SPEED BRAKE OUT
PMSG13T	CHECK ENGINES
PMSG14T	CHECK A/C ON GROUND
PMSG15T	ILLEGAL INPUT FORMAT
PMSG16T	VALUE OUT OF LIMITS
PMSG17T	INPUT STUDENT FILE DATA
PMSG18T	STUDENT INPUTS ACCEPTED ONLY DURING ZERO MODE
PMSG19T	START EXERCISE
PMSG20T	STUDENT FILE OVERFLOW
PMSG21T	NEW FILE
PMSG22T	OLD FILE
PMSG23T	ILLEGAL EXERCISE SELECTION
PMSG24T	DIFFICULTY LEVEL OUT OF LIMITS
PMSG25T	NAME NOT IN FILE
PMSG26T	FOLLOWING FILE DELETED
PMSG27T	ILLEGAL DATE FORMAT
PMSG29T	A/C ALTITUDE OUT OF LIMITS
PMSG30T	A/C HEADING OUT OF LIMITS
PMSG31T	BLANKS ARE ILLEGAL STUDENT ID
PMSG32T	RUN NUMBER XXXX
PMSG33T	LINE PRINTER MANUAL
PMSG34T	LINE PRINTER PAPER LOW
PMSGT0T	RUN TERMINATED BY
PMSGT3T	CRASH
PMSGT4T	DEPRESS THE RESET-TO-ZERO CONSOLE BUTTON

Table 2-9 **Typewriter Message Formats** (Cont)

PMSGT6T	SUCCESSFUL COMPLETION
PMSGT7T	SESSION COMPLETED
PMSGT8T	HIGHEST DIFFICULTY FACTOR ATTAINED
PMSGT9T	MAXIMUM RUNS THIS SESSION
PMSGT10T	TIME EXPIRED
PSMG36T	(OUTPUT BUFFER FOR ABSOLUTE PATCH 'PRT' COMMAND)
PMSGT20T	EXCEEDING HEADING LIMITS
PMSGT21T	EXCEEDING ALTITUDE LIMITS
PMSGT22T	EXCEEDING R/C LIMITS
PMSGT23T	EXCEEDING AIR SPEED LIMITS
PMSGT24T	EXCEEDING TURN RATE LIMITS
PMSGT25T	BANK ANGLE TOO HIGH
PMSGT26T	STARTING OUT OF LIMITS
PMSGT27T	LOST RADAR LOCK-ON

NAME HUNTLEY DATE 08/10/72

FLIGHT 0003 NST MANEUVER CLIMB 1833 RUN 0012 LAST RUN 1631

	ALTITUDE (FEET)	AIRSPEED (KNOTS)	AIRSPEED (MACH)	HEADING (DEGREES)	CLIMB RATE (FT/MIN)	TURN RATE (DEG/SEC)	YAW ANGLE (DEGREES)	ANGLE/ATTACK (UNITS)	ROLL ANGLE (DEGREES)	ROLL RATE (DEG/SEC)
ASSIGNED	20000.0		0.419	0	0	0	0	0	0	0
ENTRY	0	198.3	0.308	42.2	0	-2.460	0	13.6	0	0
			15 SECOND PARAMETER AVERAGE ERRORS							
000-015	-19860.3		-0.066	27.9	1096.6	-2.140	0.145	13.5	-24.4	-1.763
015-030	-19478.4		0.050	-1.2	2477.8	-1.116	-0.120	9.4	-16.7	2.010
030-045	-18517.6		-0.108	10.8	3746.1	-0.472	0.128	6.6	-15.8	-3.195
045-060	-17984.7		-0.127	2.9	1905.7	-1.265	-0.196	5.9	-24.6	3.907
060-075	-14435.0		-0.004	-0.5	5685.4	-0.065	0.244	5.6	-3.8	-0.559
075-090	-12985.3		-0.029	-2.3	6059.0	-0.072	0.409	4.8	-1.6	0.281
090-105	-10807.1		-0.012	-5.3	5724.9	0.040	0.367	5.3	0.4	0.248
105-120	-9045.3		0.004	-78.4	5963.4	1.564	-0.024	5.9	40.1	2.736
120-135	-7480.1		0.010	-49.6	5671.4	2.135	0.246	6.6	46.4	-0.300
135-150	-5803.5		0.005	-17.2	6031.0	2.064	0.414	6.8	42.2	-1.799
150-165	-3828.1		0.002	-1.6	6059.0	0.348	0.355	5.7	9.3	-1.650
165-180	-2059.6		0.005	-0.7	6015.6	-0.047	0.370	6.2	-2.0	0.406
180-186	-575.6		-0.002	-0.5	6059.0	0.105	0.470	6.2	1.4	0.172
FINAL	19928.0	423.2	0.893	59.2	6059.0	0.033	0.819	6.2	0.9	-2.833

RUN TERMINATED BY SUCCESSFUL COMPLETION

LEG SCORES 177.691 66.425 32.681 TOTAL SCORE 95.341
ACTUAL RUN TIME 187 SECONDS NOMINAL RUN TIME 185 SECONDS TOTAL RUNS THIS FILE 0033

FLIGHT 0003 NST MANEUVER ATTACK 1732 RUN 0013 LAST RUN 1631

	ALTITUDE (FEET)	AIRSPEED (KNOTS)	AIRSPEED (MACH)	HEADING (DEGREES)	CLIMB RATE (FT/MIN)	TURN RATE (DEG/SEC)	YAW ANGLE (DEGREES)	ANGLE/ATTACK (UNITS)	ROLL ANGLE (DEGREES)	ROLL RATE (DEG/SEC)
ASSIGNED	20000.0		0.899	180.0	0	-3.6	0	0	-60.0	0
ENTRY	20076.9	430.4	0.909	177.0	-239.1	0.341	-1.270	5.8	0.9	17.136
			15 SECOND PARAMETER AVERAGE ERRORS							
000-015	-8.5		0.002	-25.7	-920.6	0.691	-0.051	8.3	0.6	-2.980
015-030	-157.5		-0.003	-78.0	111.5	0.030	-0.390	8.8	-1.0	-0.087
030-045	-76.6		0.002	-131.0	235.5	0.398	-0.548	8.4	0.0	1.708
045-060	-51.7		0.014	28.9	-101.4	1.491	-0.824	7.7	15.7	4.370
FINAL	19849.1	445.0	0.932	352.5	-1620.4	-0.070	-0.617	4.6	22.5	-6.446

RUN TERMINATED BY SUCCESSFUL COMPLETION

GATE PENETRATION DATA:
X-DISTANCE 1.06 MI Y-DISTANCE 0.25 MI Z-DISTANCE 56 FT
LOOK-ANGLE -13 DEG HEADING 1 DEG MACH 0.91

Figure 2-19. Example of Climb, Attack, and Descent Line Printer Output (Sheet 1 of 2)

```
ATTACK FIRING DATA:
   X-DISTANCE  1.04 MI      Y-DISTANCE 0.19 MI      Z-DISTANCE   12 FT
   LOOK-ANGLE    -0 DEG     HEADING     349 DEG     MACH        0.90

      LEG SCORES    98.926                   12.004     TOTAL SCORE  110.930
      ACTUAL RUN TIME   62 SECONDS   NOMINAL RUN TIME   60 SECONDS   TOTAL RUNS THIS FILE 0034
****************************************************************************************************

      FLIGHT  0003     NST MANEUVER     DESCENT 1833     RUN 0014  LAST RUN  1732
```

	ALTITUDE (FEET)	AIRSPEED (KNOTS)	AIRSPEED (MACH)	HEADING (DEGREES)	CLIMB RATE (FT/MIN)	TURN RATE (DEG/SEC)	YAW ANGLE (DEGREES)	ANGLE/ATTACK (UNITS)	ROLL ANGLE (DEGREES)	ROLL RATE (DEG/SEC)
ASSIGNED	20000.0	280.0		330.0	0	0	0	0	0	0
ENTRY	19793.7	446.3	0.932	359.6	-378.7	1.095	0.928	6.0	10.6	-21.191
			15 SECOND PARAMETER AVERAGE ERRORS							
000-015	-275.0	147.7		17.7	-467.5	-1.701	0.118	7.3	-40.3	-3.905
015-030	-0.1	79.8		-3.4	2172.4	-0.465	-1.186	7.4	-9.1	3.258
030-045	107.1	49.3		-0.6	-1050.7	0.187	-0.485	7.7	1.7	-0.030
045-060	6067.7	18.3		21.7	-218.5	0.120	-0.105	8.5	0.5	-1.961
060-075	17814.2	-6.6		55.9	-4132.8	-0.746	-0.493	7.1	-29.9	0.172
075-090	15550.1	18.0		41.6	-6059.0	-0.945	-0.235	8.7	-19.5	-0.739
090-105	13849.2	15.6		16.5	-5323.1	-2.125	-0.467	9.7	-37.4	0.033
105-120	12696.9	2.8		0.6	-4017.1	-0.278	-0.412	8.0	-7.1	1.845
120-135	11259.1	7.2		1.1	-6056.8	-0.026	-0.223	8.1	-3.1	0.142
135-150	9541.4	16.7		-0.9	-5421.5	-0.124	-0.129	8.2	-3.2	-0.163
150-165	8375.3	43.7		-102.2	-3938.2	-1.029	-0.216	8.9	-18.2	-1.893
165-180	7796.5	32.4		-132.5	-1181.1	-1.853	-0.495	10.2	-30.2	-0.275
180-195	7515.3	4.7		162.5	-1660.1	-1.886	-0.671	11.0	-31.3	0.365
195-210	6705.3	2.6		134.1	-4265.8	-1.979	-0.626	11.3	-31.3	-0.130
210-225	5448.2	13.6		103.2	-4806.0	-2.134	-0.607	11.5	-30.8	-0.175
225-240	4521.0	8.7		69.3	-3112.8	-2.178	-0.666	11.3	-32.8	-0.002
240-255	3621.0	8.0		38.4	-3933.7	-2.100	-0.661	11.6	-30.2	0.160
255-270	2824.0	2.3		7.2	-2907.6	-1.820	-0.845	11.0	-29.5	1.050
270-285	1673.2	18.3		0.4	-5511.3	0.192	-0.592	10.5	1.5	0.741
285-300	729.4	17.2		0.2	-2097.6	-0.051	-0.290	10.2	-2.8	-0.015
300-313	346.9	2.7		0.3	-1975.8	0.025	-0.324	10.0	-1.6	0.145
FINAL	1536.5	253.8	0.400	45.5	-3319.3	0.175	-0.445	9.4	-0.9	0.703

```
      RUN TERMINATED BY   SUCCESSFUL COMPLETION

      LEG SCORES   180.540     115.657     100.693     TOTAL SCORE  208.146
      ACTUAL RUN TIME  314 SECONDS   NOMINAL RUN TIME  280 SECONDS   TOTAL RUNS THIS FILE 0035
****************************************************************************************************

      FLIGHT  0003     NST MANEUVER     CLIMB   1833     RUN 0015  LAST RUN  1833
```

	ALTITUDE (FEET)	AIRSPEED (KNOTS)	AIRSPEED (MACH)	HEADING (DEGREES)	CLIMB RATE (FT/MIN)	TURN RATE (DEG/SEC)	YAW ANGLE (DEGREES)	ANGLE/ATTACK (UNITS)	ROLL ANGLE (DEGREES)	ROLL RATE (DEG/SEC)
ASSIGNED	20000.0		0.419	0	0	0	0	0	0	0
ENTRY										

Figure 2-19. Example of Climb, Attack, and Descent Line Printer Output (Sheet 2 of 2)

A comprehensive description of student file inputs is contained in Section IV of Appendix A – F-4/ATE/AFT PROGRAM OPERATING INSTRUCTIONS. A brief description of each student file input command follows:

$	Type out the current student file data
$FILE	Opens new file for new subject or re-opens old file for previous subject
$DATE	Updates the month, day, and year for files
$EXER	Overrides program selection of next exercise and difficulty level for the current file.
$RNUM	Overrides program selection of next run number
$DELE	Deletes specified student file.
$GO	Mandatory command used to start the next session
$SNUM	Overrides program selection of next session number

Student file data can only be input when the RESET-TO-ZERO button (zero mode) on the Monitor Console has been depressed.

All exercise parameter change commands are identified to the program by the character * (asterisk). The following exercise parameter changes are currently available to the ATE/AFT Program (GCA/EMR exercises only).

*RUN	Changes the runway orientation for subsequent runs
*WND	Changes the wind velocity parallel to the runway orientation for the next run only.

Absolute program patch commands permit the printing or modification of any memory location in the F-4/ATE/AFT program and are identified by the character, + (plus). The following absolute program patch commands are currently available to the ATE/AFT program:

+PRT	Type the contents of the indicated memory location
+MOD	Modify the indicated memory location with the value supplied

An example of the student file format and the typewriter printout associated with this file is depicted in figure 2-20. The symbols listed in the student file format contain the following data.

FNAME	Subjects name – up to eight EBCDIC characters input by $FILE command.
FDATE	Current month, day, and year – eight EBCDIC characters input by $DATE command.
FEXER	Current exercise for subject – usually determined by Adaptive Logic module but may be input by $EXER command.
FLEVL	Current difficulty level for the next run – usually determined by Adaptive Logic module but may be input by $EXER command.
FSESS	Current session number – set to 1 by the $FILE command if new file or incremented by the $FILE command if old file.
FRUN	Current run number for this session – usually controlled by the Terminate Exercise module but may be input by $RNUM command.
FTOTL	Total runs this file – set to 0 by the $FILE command if new file. Incremented by the Terminate Exercise module.
SSTIME	(Not printed.) Session start time is the F-4 cycle counter reading when a file is opened or reopened ($FILE command). It is used primarily to control the session length.
FDFTABX	(Not printed.) Current difficulty table index which is used in conjunction with DFTABXM1 to determine difficultly level increment for next run.
DFTABXM1	(Not printed.) Difficulty table index of previous run which is used in conjunction with FDFTABX to determine difficulty level increment for next run.
LRUNSAV	Exercise level for the last run made by the subject.
FLAT	(Not printed.) Difficulty level for last NST-ATTACK run made by the subject.

FNAME	D1		D6		C8		D5		EBCDIC	(JOHN
	E2		D6		D5		40		EBCDIC	SON)
FDATE	F1		F0		61		F2		EBCDIC	(10/2
	F8		61		F7		F2		EBCDIC	8/72)
FEXER	D5		E2		E3		40		EBCDIC	(NST)
FLEVL	0D		0A		0B		00		BYTE	(131011)
									INTEGER	
FSESS	0	0	0	0	0	0	0	4	INTEGER	(4)
FRUN	0	0	0	0	0	0	0	3	INTEGER	(3)
FTOTL	0	0	0	0	0	0	3	7	INTEGER	(37)
SSTIME	0	0	0	0	1	F	C	4	INTEGER	(1FC4)
FDFTABX	0	0	0	0	0	0	0	2	INTEGER	(2)
DFTABXM1	F	F	F	F	F	F	F	F	INTEGER	(-1)
LRUNSAV	—		—		—		—		(NOT USED BY NAV/STRIKE)	
FLAT	FO		F8		F1		F2		EBCDIC	(0812)
FLCL	F1		F4		F2		F2		EBCDIC	(1422)
FLDE	F1		F0		F2		F3		EBCDIC	(1023)

Student File Format

NAME	JOHNSON	DATE	10/28/72
EXER	NST	LEVEL	CL13AT10DE11
SESS	0004	RUN	0003
TOTAL RUNS THIS FILE			0037

Typewriter Printout

Figure 2-20. **Example of Student File Format and Typewriter Printout**

AD-A104 035 LOGICON INC SAN DIEGO CA F/G 5/9
AFT PROGRAM DESCRIPTION NAVIGATION/STRIKE TASKS. PHASF II.(U)
SEP 72 R M JOHNSON N61339-72-C-0108
UNCLASSIFIED SDR-120 NL

2 OF 3
AD
A104 035

FLCL	(Not printed.) Difficulty level for last NST-CLIMB run made by the subject.
FLDE	(Not printed.) Difficulty level for last NST-DESCENT run made by the subject.

e. Inputs

1. Internal Inputs: ALOGPTR — Pointers to Exercise Difficulty Level Tables

2. External Inputs:

CLOCK	F-4 Cycle Counter
T99DI3	Discrete Input Word #3

3. Constants:

BLEVL	Basic Difficulty Level
CTAB	BCD to Hexadecimal Conversion Table
DATETAB	$DATE Input Limits
DFEMR	EMR Exercise Difficulty Level Limits
DFGCA	GCA Exercise Difficulty Level Limits
DFLAG	$DELE Printout Flag
DFTAB	Pointer to Current Difficulty Level Table
EPTAB	Legal Exercise Parameters Table
FDATE	Student File – Date
FNAME	Student File – Name
HEXTAB	Hexadecimal to BCD Conversion Table
IODKEY	Keyboard Input Buffer Doubleword
KEYBUF	Keyboard Input Buffer
LMSGHEAD	Head of Line Printer Message Queue

LMSG(XX) or ILMSG(XX)	I/O Command Doubleword for Line Printer Messages LMSG1T – LMSG27T
LMSG(XX)T or ILMSG(XX)T	ATE Line Printer Messages
LPFTM	Line Printer – 1st Time Manual Switch
LPFTPL	Line Printer – 1st Time Paper Low Switch
NRUNWAY	New Runway Orientation
NVWAR	New Velocity of Wind Along Runway
PDATE	Print-out Buffer – Date Parameter
PEXER	Print-out Buffer – Exercise Parameter
PLEVL	Print-out Buffer – Difficulty Level Parameter
PMSGHEAD	Head of Typewriter Message Queue
PMSG(XX)	I/O Command Doublewords for Typewriter Messages PMSG1T – PMSG36T
PMSGT(XX)	I/O Command Doublewords for Typewriter Messages PMSGT1T – PMSGT27T
PMSG(XX)T	ATE/AFT Typewriter Messages
PMSGT(XX)T	ATE/AFT Typewriter Messages
PNAME	Print-out Buffer – Name Parameter
PRUN	Print-out Buffer – Run Number Parameter
PSESS	Print-out Buffer – Session Number Parameter
PTOTL	Print-out Buffer – Total Runs Parameter
SFITAB	Legal Student File Input Commands
TYPESW	Typewriter in Use Switch
VALPTR	Pointers to Exercise Parameters Location

f. Outputs

1. Internal Outputs:

ALOGTAB	Pointer to Current Difficulty Level Tables
CEXER	Current Exercise Number
DFTABX	Current Difficulty Level Table Index
DFTABXM1	Previous Difficulty Level Table Index
EXTAB	Legal Exercises Table
FDFTABX	Student File – Current Difficulty Level Table Index
FEXER	Student File – Exercise
FLAT	Student File – Last Attack Run Status
FLCL	Student File – Last Climb Run Status
FLDE	Student File – Last Descent Run Status
FLEVL	Student File – Difficulty Level
FRUN	Student File – Run Number
FSESS	Student File – Session Number
FTOTL	Student File – Total Runs
GOFLAG	Exercise Start Flag
KEYSW	Keyboard in Use Flag
LRUNSAV	Exercise/Level for Last Run
ROUTE	Program Route Value
SFNUM	Current Student File Number
SSTIME	Student File – Session Start Time

2. External Outputs: None

g. Program Entrances

1.	BAL, 15 DATA	POUT:1 WA(PMSG)	Typewriter Message Queue Routine
2.	BAL, 15 DATA	LPOUT:1 WA(LPMSG)	Line Printer Message Queue Routine
3.	BAL, 15 DATA	PMSG:1	Typewriter Output Routine
4.	BAL, 15 DATA	LPMSG:1	Line Printer Output Routine
5.	BAL, 15 DATA	KEYIN:1	Keyboard Input Routine

h. Exits

1.	B	*15	(calling location +2) POUT:1 Exit
2.	B	*15	(calling location +2) LPOUT:1 Exit
3.	B	*PMSGEX	(calling location +1) PMSG:1 Exit
4.	B	*LPMSGEX	(calling location +1) LPMSG:1 Exit
5.	B	*KEYEX	(calling location +1) KEYIN:1 Exit

i. Subroutines Called

1.	CTOH	Convert EBCDIC to Hexadecimal
2.	HEXTOBCD	Convert Hexadecimal to EBCDIC
3.	NPACK	Pack Input Characters into Buffer
4.	PRINTF	Print Current Student File Data

j. Memory Requirements

1.	Instructions	679
2.	Data	304

k. Type of Program Module. ATE/AFT Background Programs

l. Flow Charts. See figures 2-21 through 2-26.

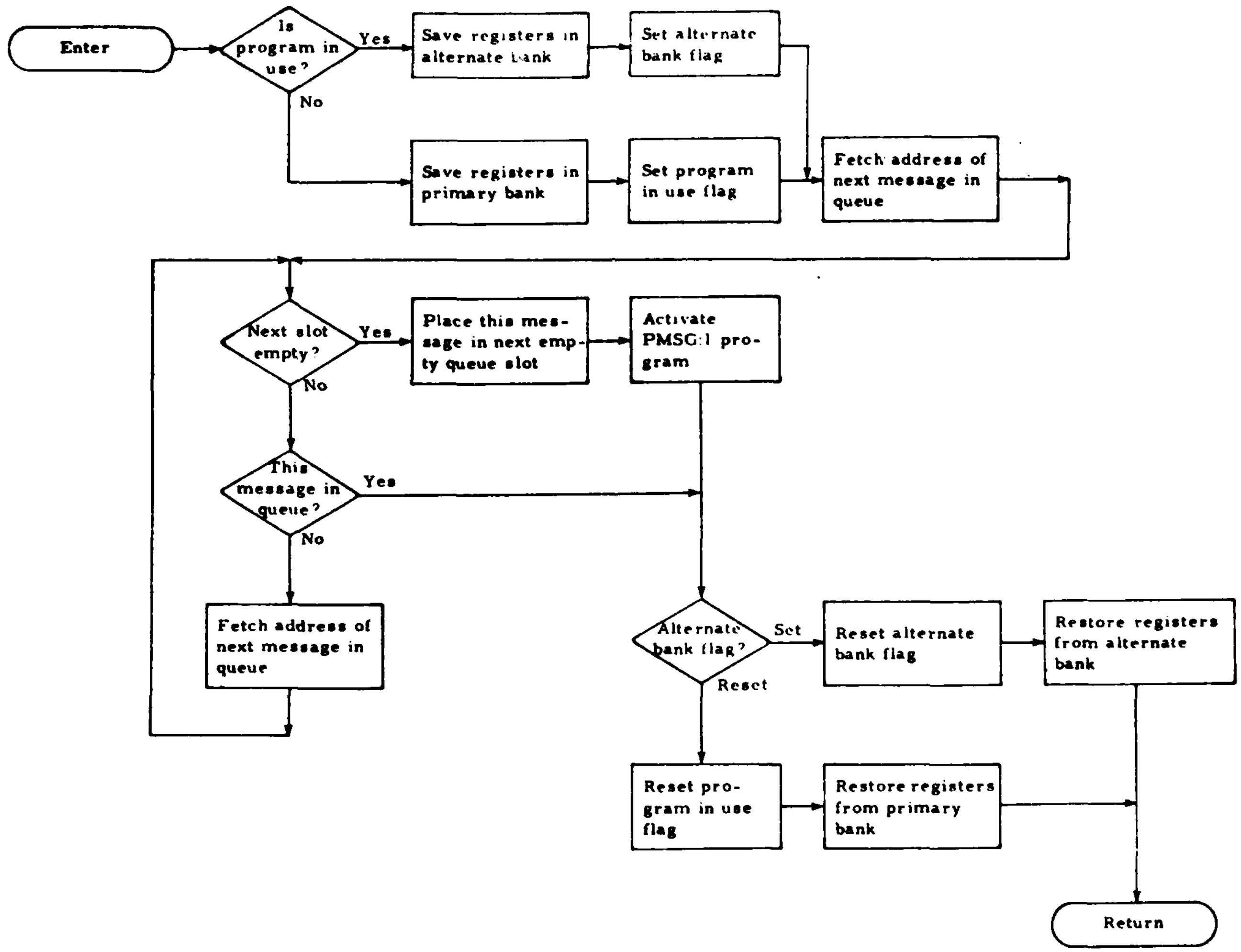


Figure 2-21. Data Input/Output – POUT:1 Subroutine

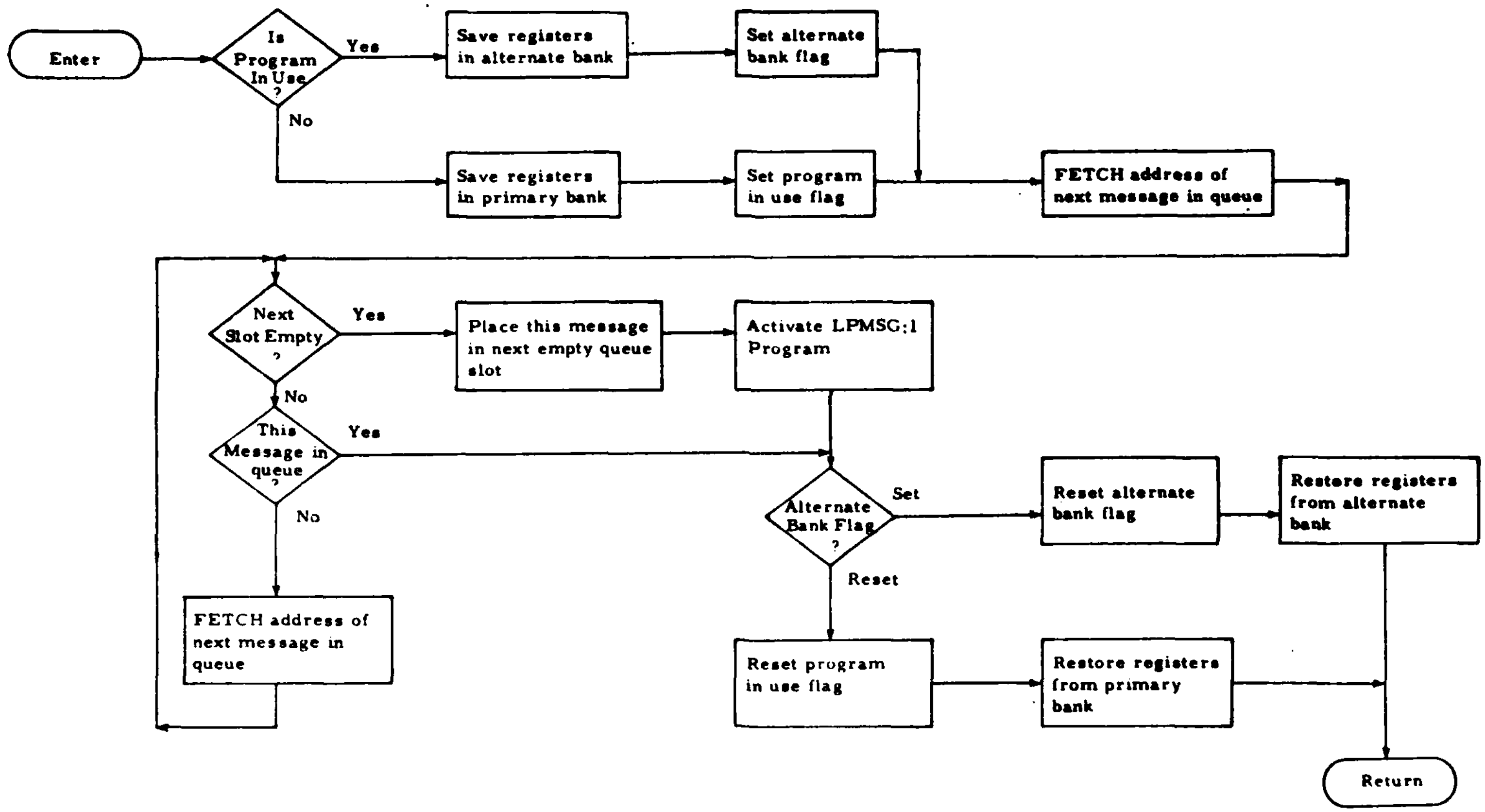


Figure 2-22. Data Input/Output — LPOUT:1 Subroutine

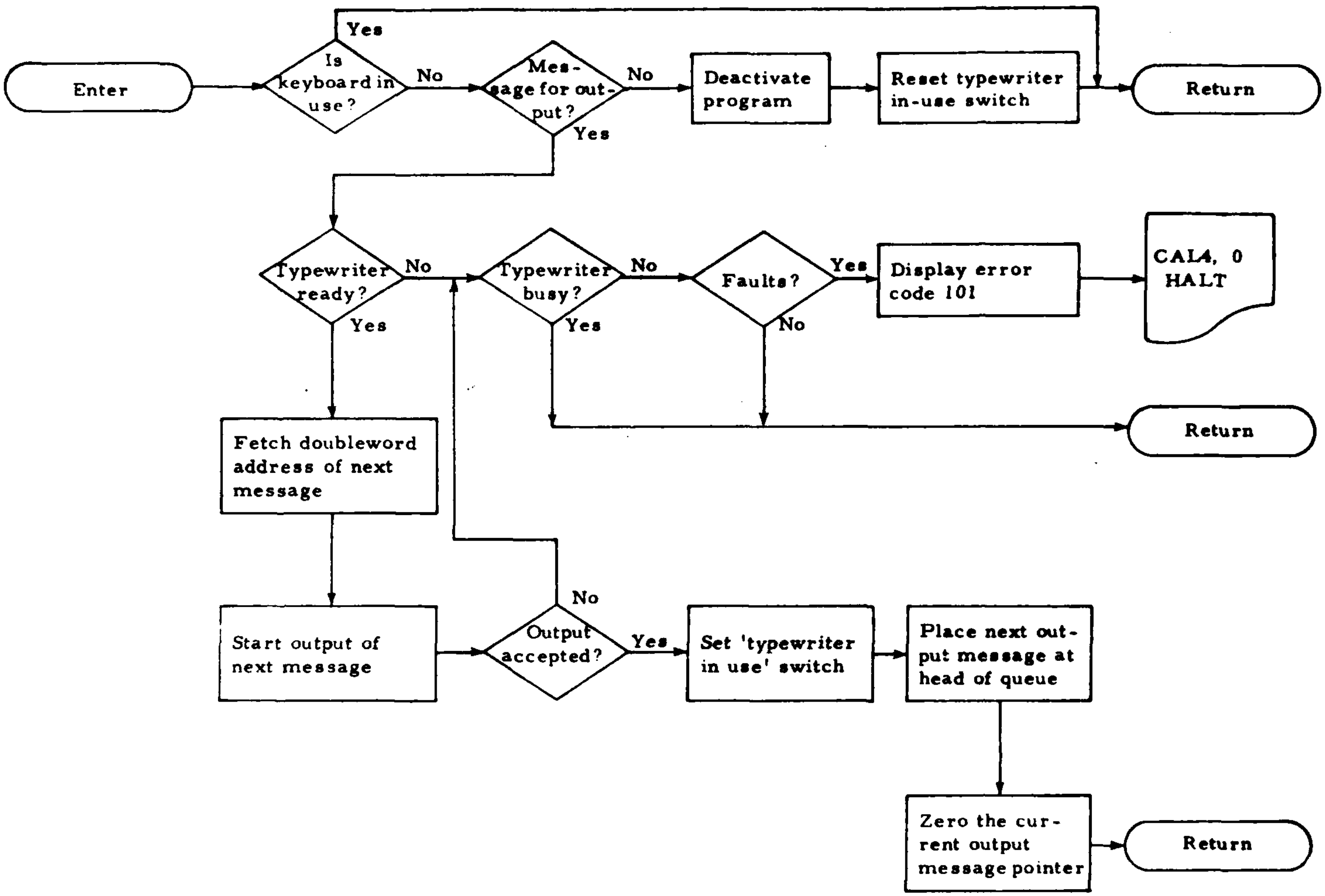


Figure 2-23. Data Input/Output – PMSG:1 Subroutine

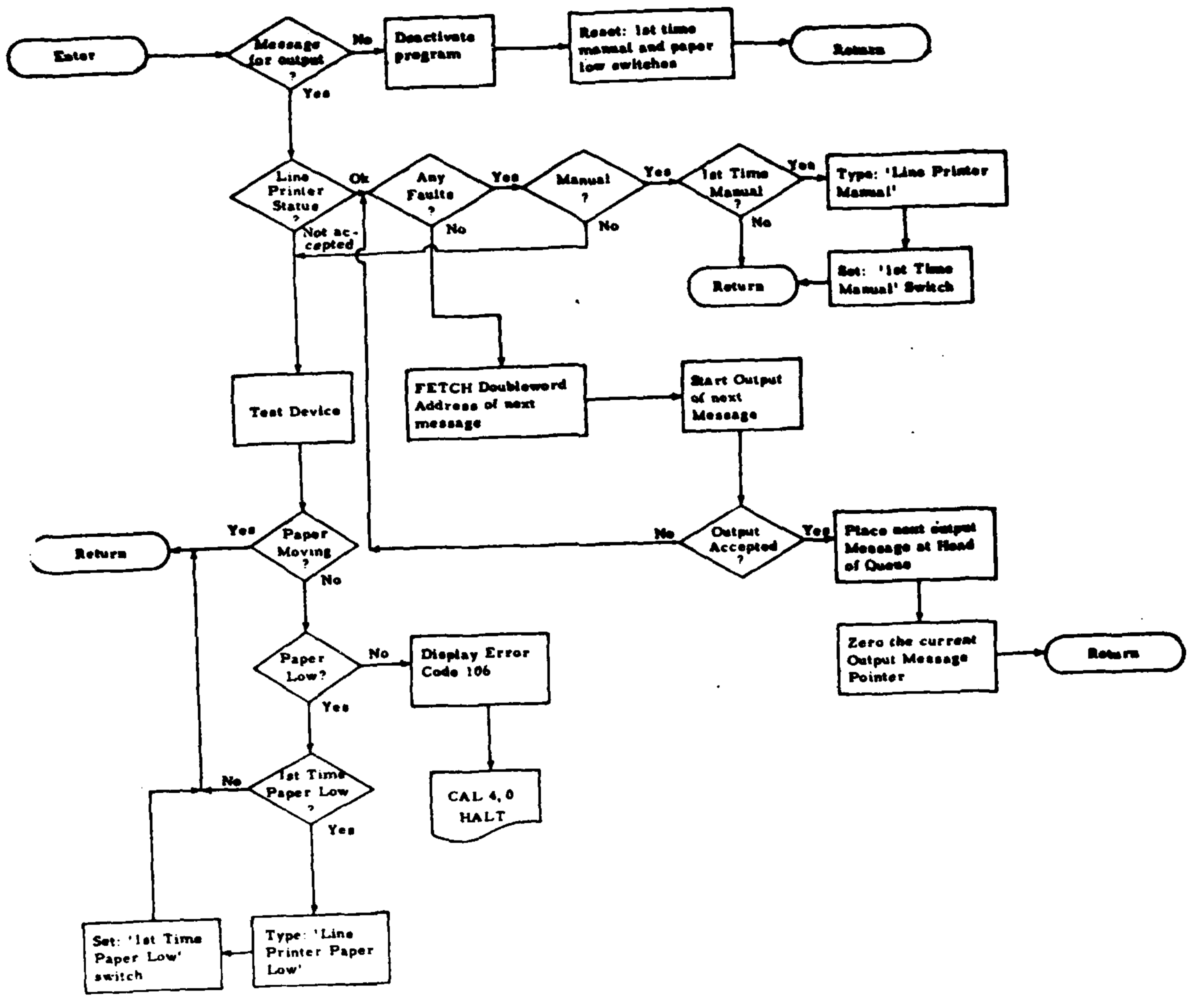


Figure 2-24. Data Input/Output – LPMSG:1 Subroutine

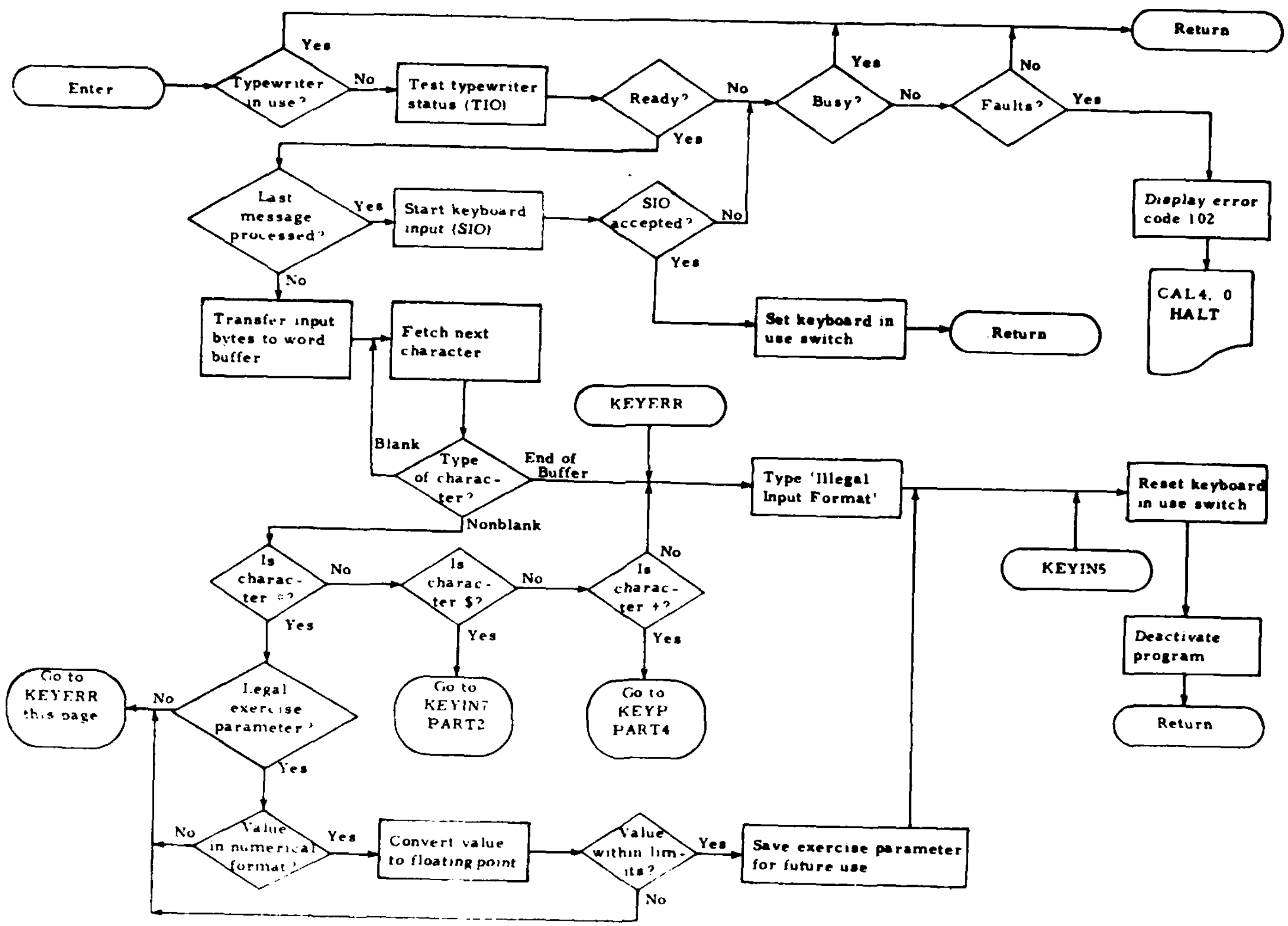


Figure 2-25. Data Input/Output – KEYIN:1 Subroutine (Sheet 1 of 5)

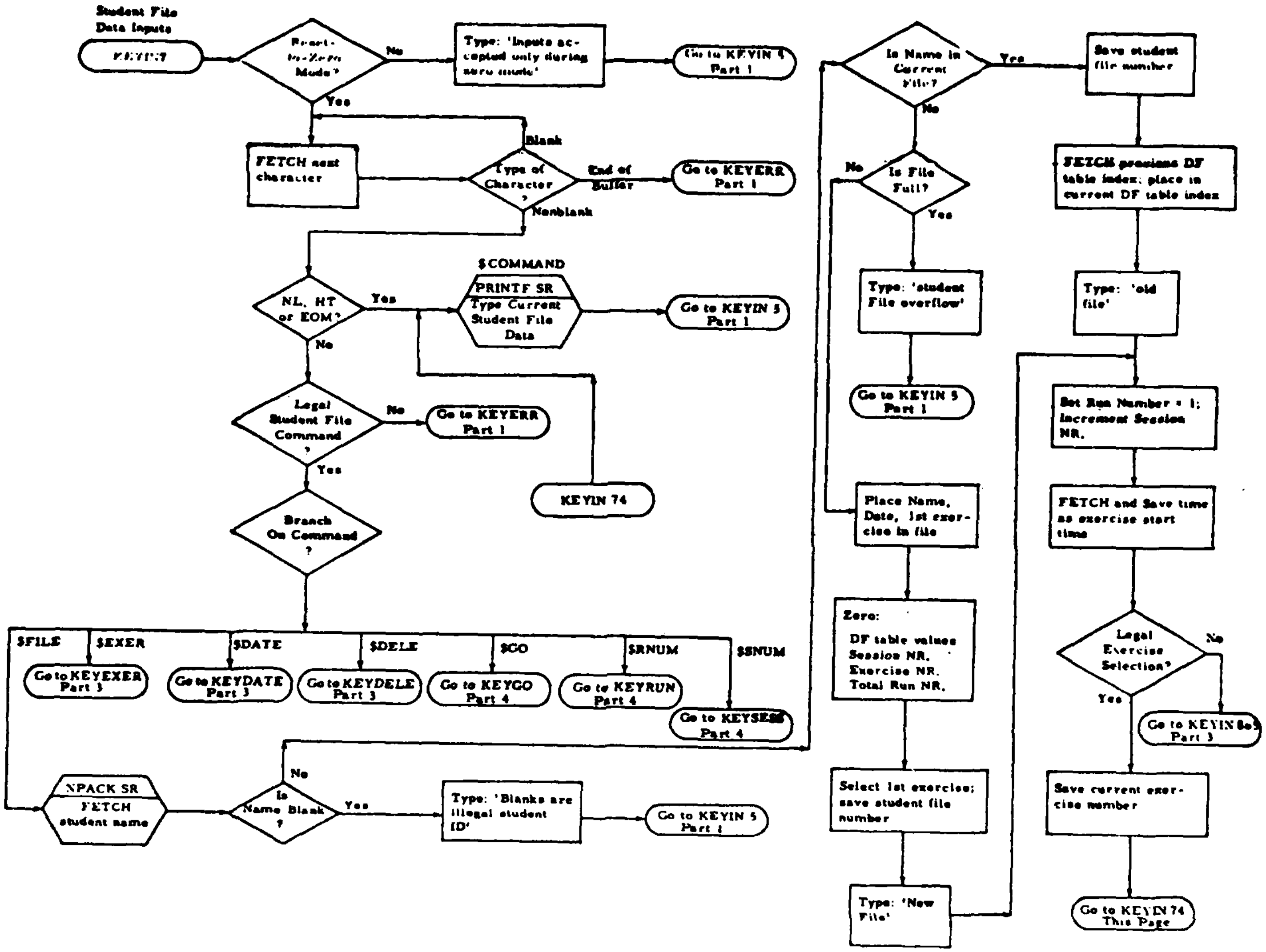


Figure 2-25. Data Input/Output – KEYIN:1 Subroutine (Sheet 2 of 5)

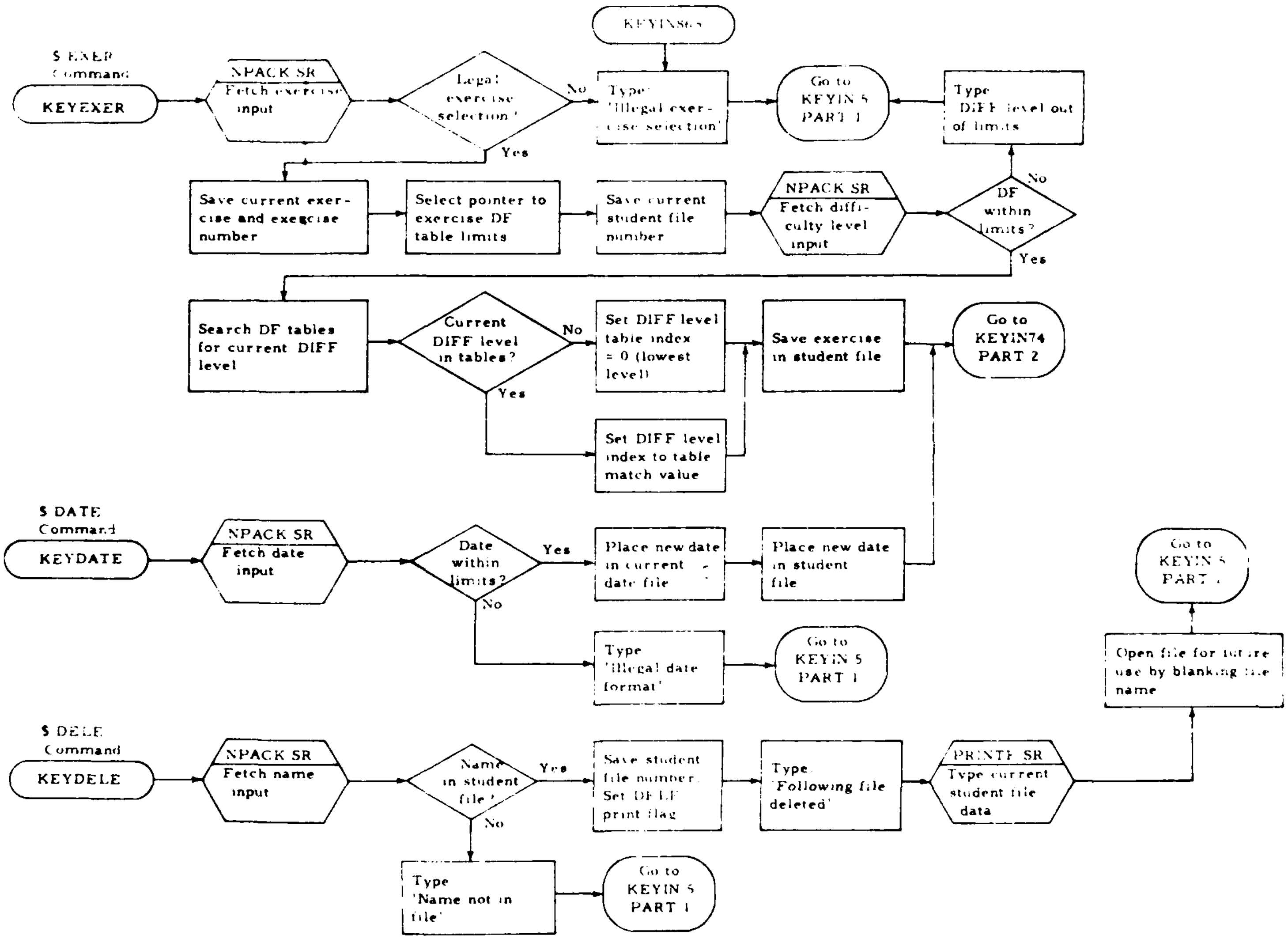


Figure 2-25. Data Input/Output – KEYIN:1 Subroutine (Sheet 3 of 5)

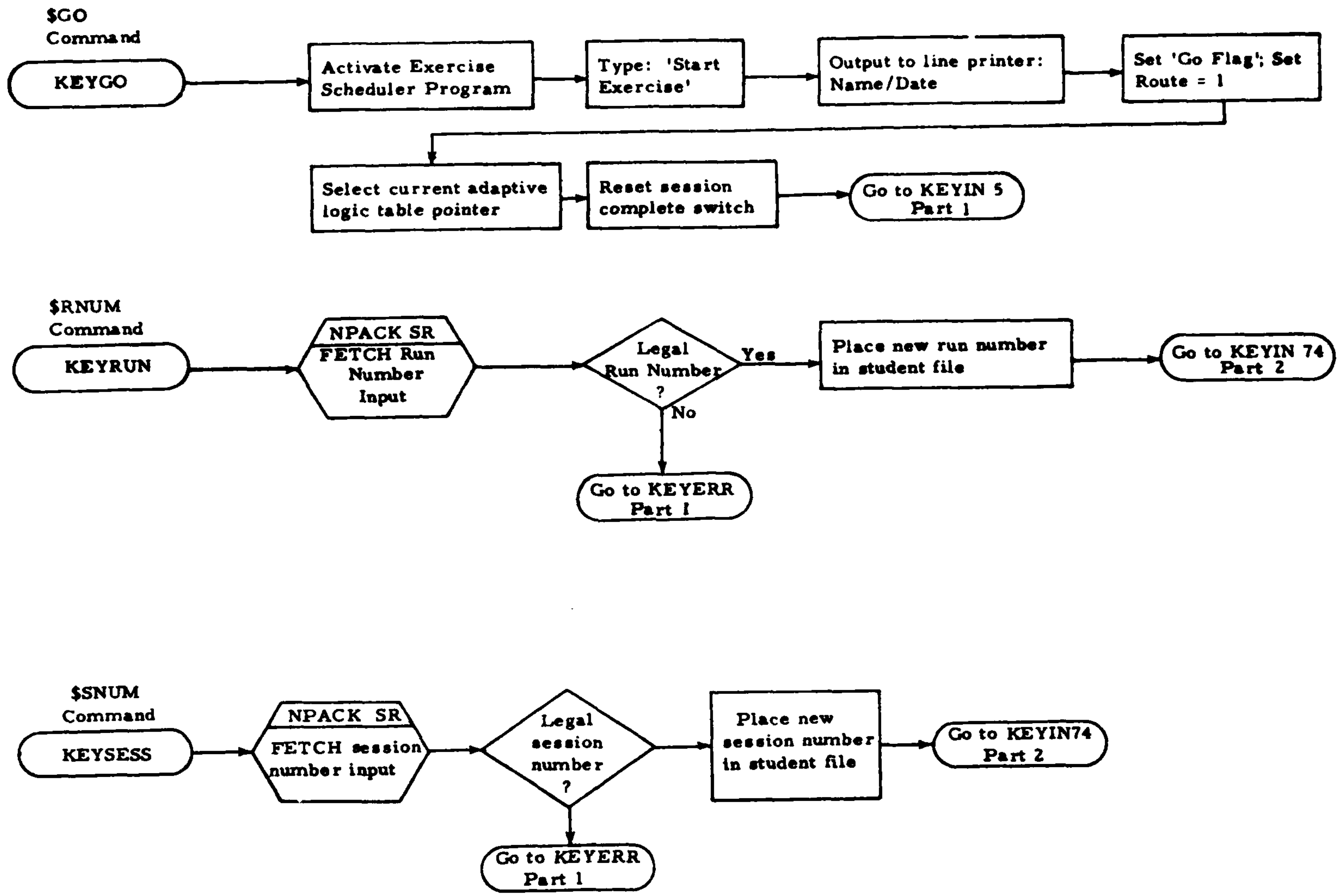


Figure 2-25. Data Input/Output — KEYIN:1 Subroutine (Sheet 4 of 5)

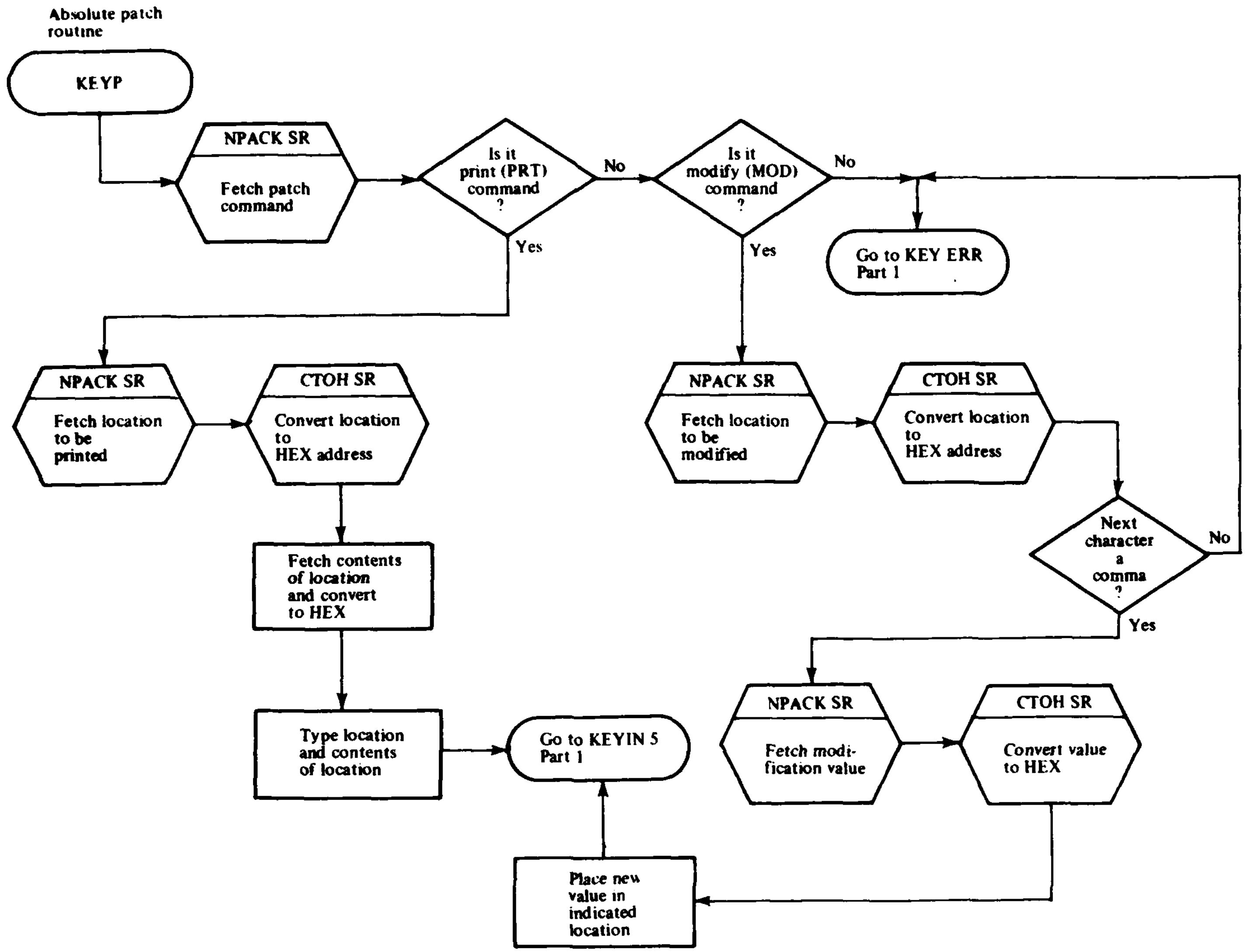


Figure 2-25 Data Input/Output — KEYIN:1 Subroutine (Sheet 5 of 5)

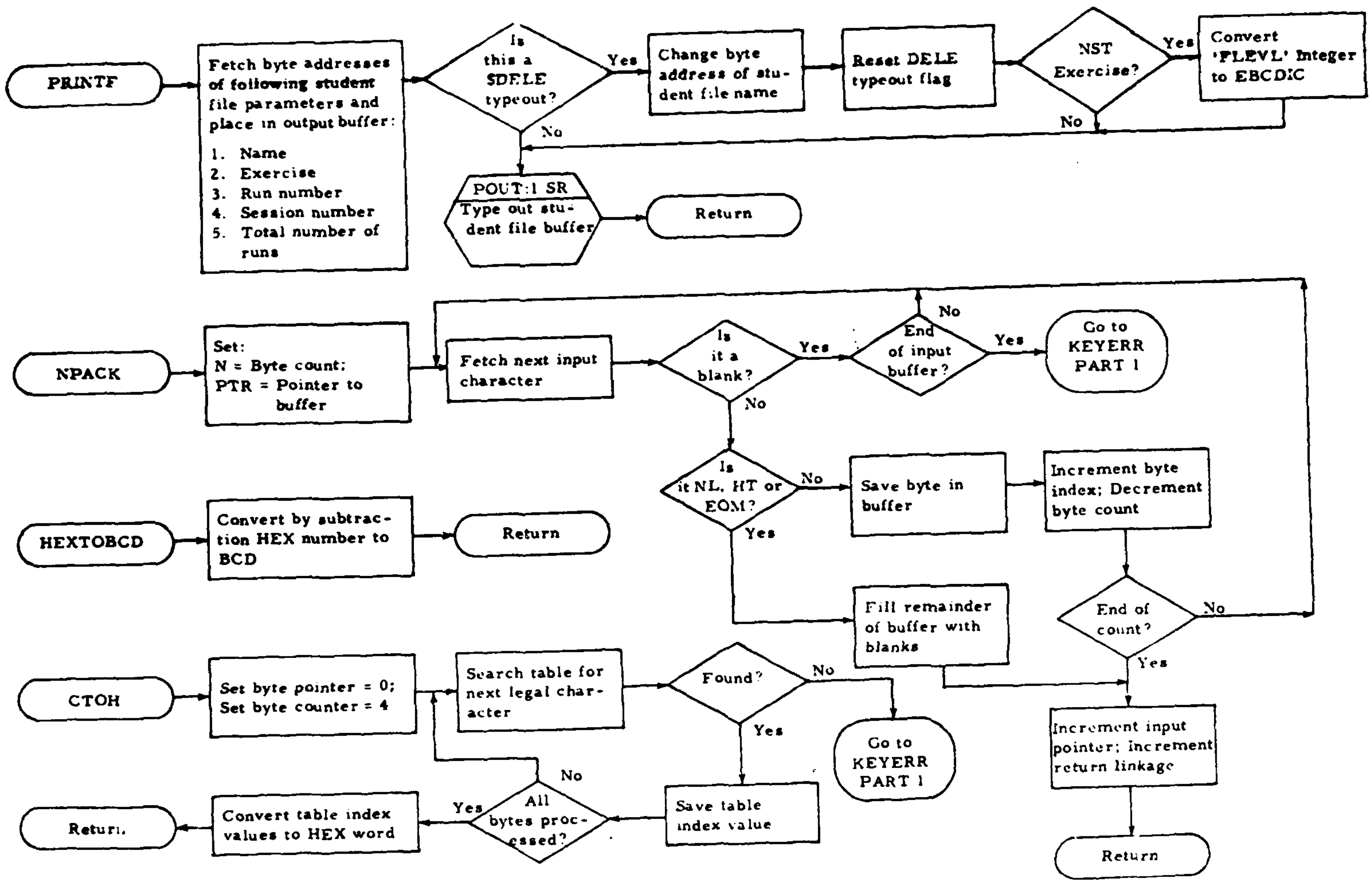


Figure 2-26. Data Input/Output — Miscellaneous Subroutines

2.14 EXERCISE SCHEDULER

a. Program Module Name. Exercise Scheduler (EXSC)

b. Purpose. The purpose of the EXSC module is to determine which ATE/AFT exercise has been selected and to set up the proper program linkages for its execution.

c. Requirements. The EXSC module is required to:

1. Determine which ATE/AFT exercise has been selected via the typewriter input or the Adaptive Logic module.

2. Set up the proper program linkages for the selected exercise

3. Initialize selected exercise parameters

4. Activate the correct program modules to successfully monitor the selected exercise

5. Select the difficulty level for the current exercise.

d. Description. The EXSC module may be activated by one of the following methods:

1. Input through the typewriter.

2. Internally by the ATE Program upon completion of a particular run (if not the last run of a session).

Entry by the first method requires that the exercise be provided by the operator, while the second method is determined by the program automatically.

Currently, the four exercises available are: the GCA Approach, Emergency Procedures, Instrument Flight Maneuvers, and Navigation/Strike Maneuvers. The program is designed to permit the incorporation of additional exercises, as desired, with minimum program changes.

In addition to the selection of the exercise and the initialization of certain parameters for the exercise, the EXSC module also selects the difficulty levels to be used for the next run. This is done by calling either the DFSELECT subroutine contained in the ALOG program module or the NSTS subroutine for NAV/STRIKE maneuvers.

e. <u>Inputs</u>

1. Internal Inputs:

CMHEAD	Head of COGNITRONICS Queue
CRASH	F-4 Crash Switch
EXTAB	Legal Exercises Table
FEXER	Student File – Exercise
FLAT	Student File – Last Attack Run Status
FLCL	Student File – Last Climb Run Status
FLDE	Student File – Last Descent Run Status
FLEVL	Student File – Difficulty Level
FRUN	Student File – Run Number
FSESS	Student File – Session Number
ROUTE	ATE/AFT Program Route Value
SBOVR	Speed Brake Override Switch
SCSW	Flight Complete Switch
SFNUM	Current Student File Number
TERMCODE	Run Termination Code
TRSKIP	Bypass Switch for Data Processing and Adaptive Logic

2. External Inputs:

CLOCK	F-4 Cycle Counter
CRASH	F-4 Crash Flag
T99D03	F-4 Digital Output Word #3
VI	F-4 Indicated Airspeed

3. Constants:

ABMASK	Aircraft-on-Ground Bit Mask
CMSGFC	'Flight Complete' Voice Message
CMSGLU	'Landing Underway" Voice Message
EXTAB	Table of Legal Exercises
IOCDM2	Doubleword to Activate COGNITRONICS

f. <u>Outputs</u>

1. Internal Outputs:

CEXER	Current Exercise Number
CLPHASE	Climb Monitor Phase
CMSWORD	COGNITRONICS Output Address
DEPHASE	Descent Monitor Phase
EMPH	Emergency Processor Program Phase
IFMINP	IFM Initialize Phase
ILMSG2T	Line Printer Message
LMSG2T	Line Printer Message
PABP	Pre-Airborne Program Phase
PGCAP	Pre-GCA Program Phase
PGCA5TIM	Pre-GCA Program Phase 5 Time
PMSG23T	Typewriter Message
PMSG32T	Typewriter Message
STPHASE	Attack Monitor Phase
TOPHASE	Pre-Take Off Phase

2. External Outputs: None

g. Program Entrances. BAL, 15 EXSCH:1

h. Exits. B *EXSAVE

i. Subroutines Called

1. ACTLOG Activate COGNITRONICS

2. DFSELECT Select Difficulty Factors

3. HEXTOBCD Convert Hexadecimal to EBCDIC

4. POUT:1 Typewriter Message Output

5. NSTS Navigation/Strike Task Selector

6. QINSERT COGNITRONICS Message Output

j. Memory Requirements

1. Instructions 200

2. Data 24

k. Type of Program Module. Background Program

l. Flow Charts. See figures 2-27 and 2-28.

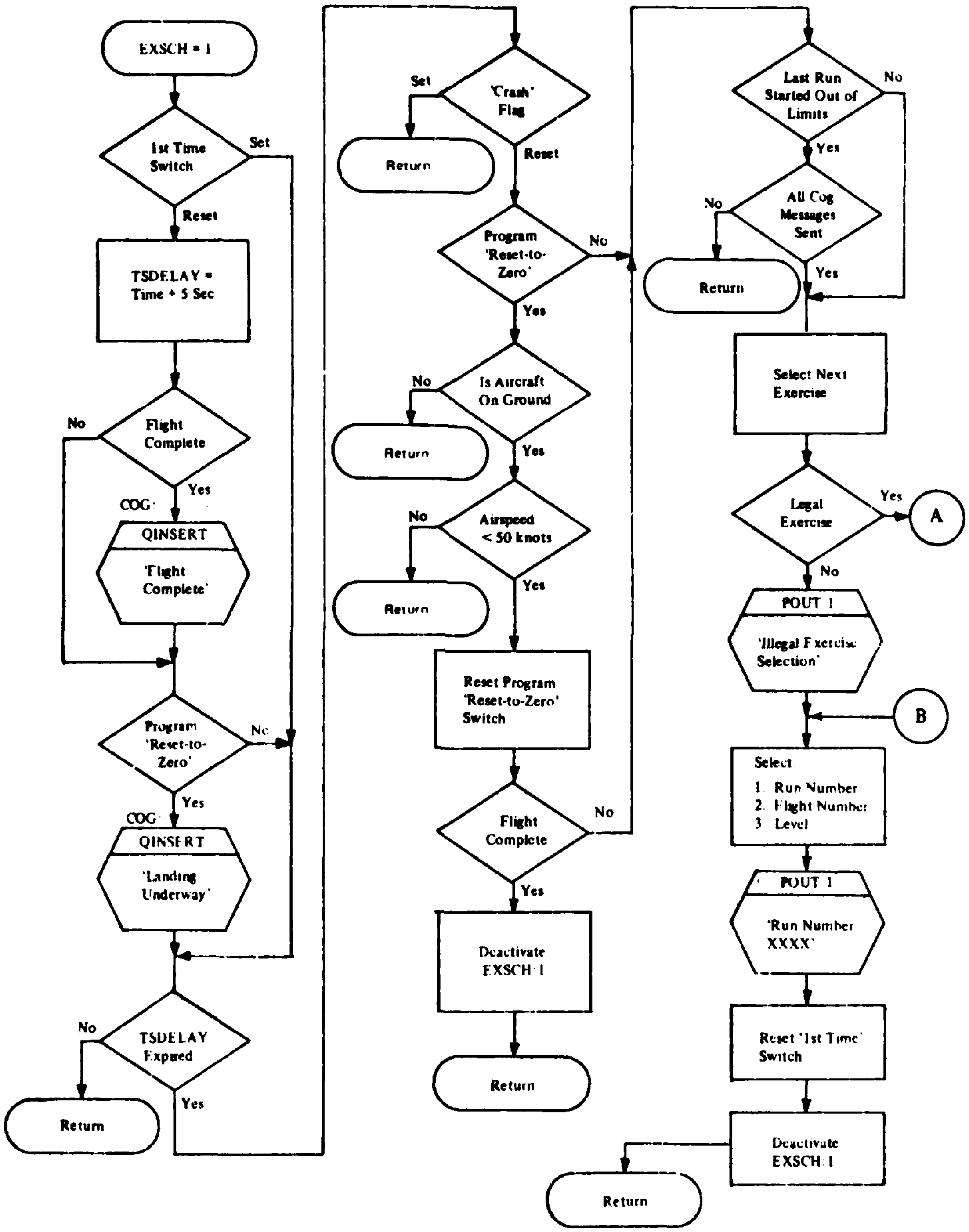


Figure 2-27. Exercise Scheduler (Sheet 1 of 3)

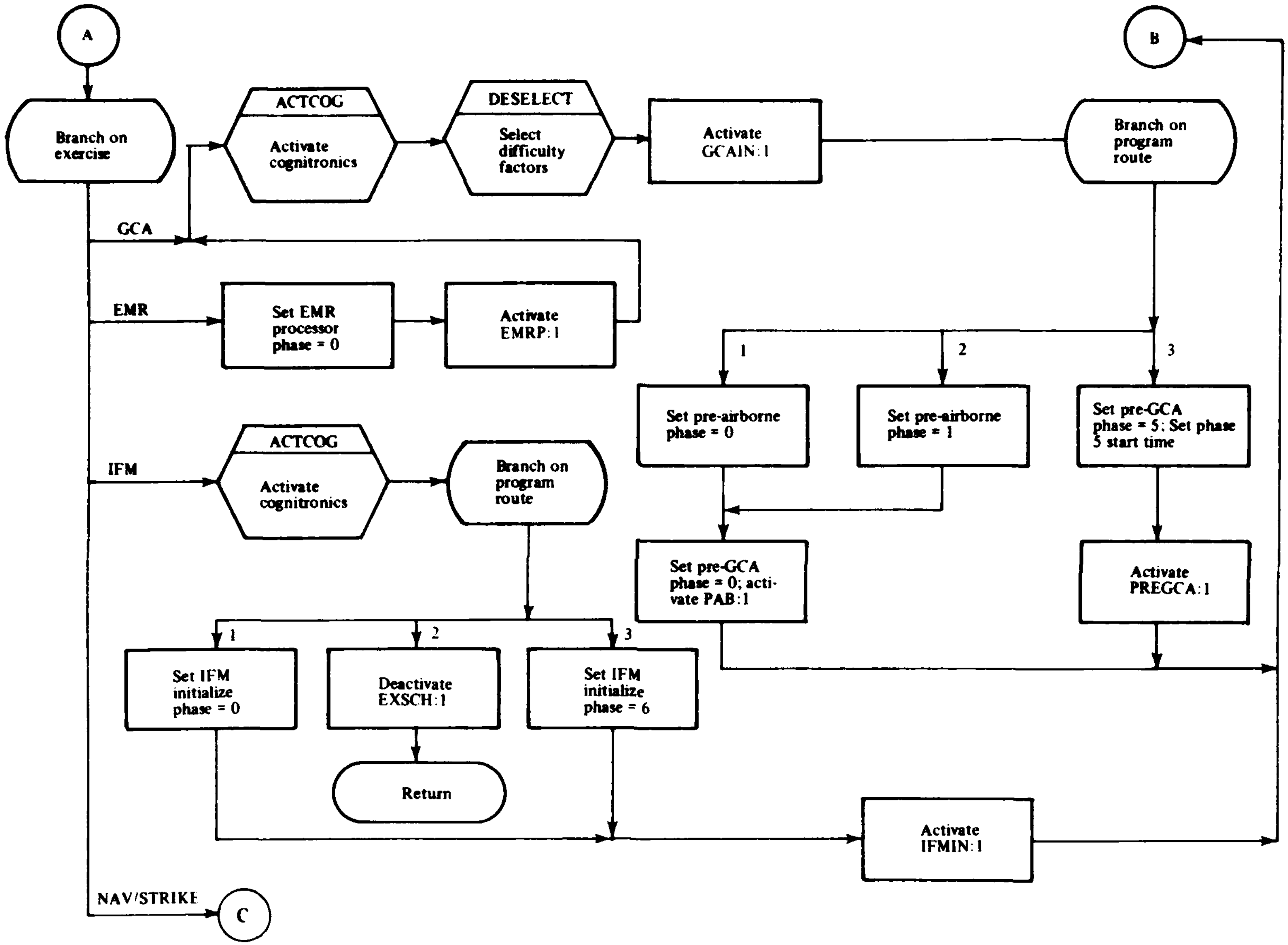


Figure 2-27. Exercise Scheduler (Sheet 2 of 3)

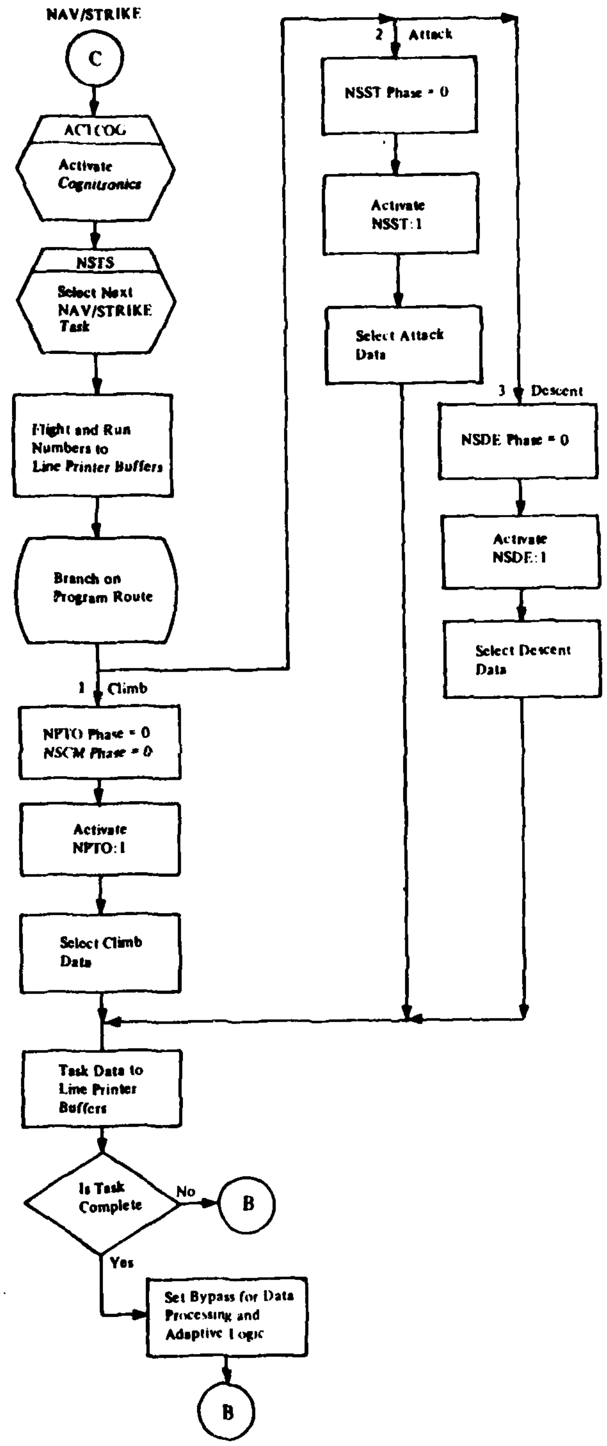


Figure 2-27. Exercise Scheduler (Sheet 3 of 3)

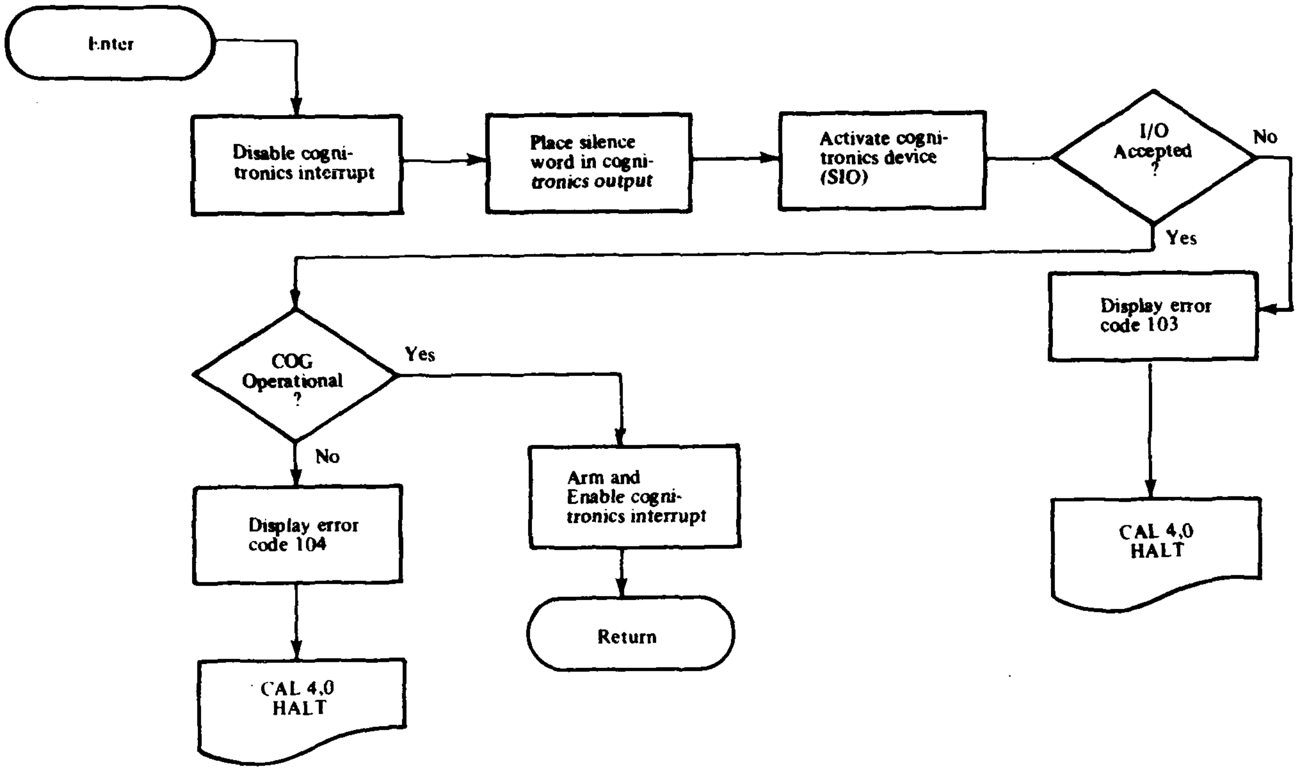


Figure 2-28. Activate COGNITRONICS (ACTCOG) Subroutine

2.15 CONTROL AIRCRAFT TO BASIC IFM/NST CONFIGURATION

a. Program Module Name. Control Aircraft to Basic IFM/NST Configuration (CIFC)

b. Purpose. The purpose of the CIFC module is to monitor and control the aircraft, through the updating of heading, altitude, and airspeed, to the specified IFM/NST configuration.

c. Requirements. The CIFC module is required to:

1. Update heading to the specified value.

2. Update and lock altitude to the specified value.

3. Update and lock mach or airspeed to the specified value.

4. Check the desired conditions for wheels, speed brake, etc. and output appropriate messages to the COGNITRONICS if the conditions are not met.

5. Determine if the aircraft is within the specified parameter limits established for the basic IFM/NST configuration.

d. Description. The CIFC module is composed of three different routines. BIFMCK and CFGCHECK are monitor routines and CTO:2 is a monitor and control routine. The BIFMCK subroutine uses a parameter limit table to determine if the aircraft is within the limits specified in the table. The BIFMCK subroutine is normally called to determine when the aircraft is in a stable enough condition to start the run.

The CFGCHECK subroutine checks the desired status of the discrete output, T99D03, for the condition of the wheels and flaps. Subroutine CFGCHECK, like subroutine BIFMCK, is called to determine if the aircraft is in the proper flight configuration. The CIFC module accomplishes its function in three separate phases. Phase 0 is concerned with updating the heading. Phase 1 is cycled through until the specified altitude is attained. At this point, the altitude is locked and Phase 2 is entered. Phase 2 then updates the velocity of the aircraft either in the form of desired airspeed or desired mach. Once all of the above three values are attained, the CIFC module is exited.

e. Inputs

1. Internal Inputs:

CBIT	Configuration bits to be checked in T99D03
CIFP	CIF Phase
DAFTALT	Desired Altitude
DAFTAS	Desired Airspeed
FCONT3	Flight Control Word

2. External Inputs:

ALT	Aircraft Altitude
CLOCK	F-4 Cycle Time
MACH	Aircraft Mach Number
VI	Indicated Airspeed

3. Constants:

DALTLIM	Desired Altitude Limit
DASLIM	Desired Airspeed Limit
DMLIM	Desired Mach Limit

f. Outputs

1. Internal Outputs:

CIFP	CIF Phase
DMACHVAL	Desired Mach Value
FCONT3	Flight Control Word
MFTIME	Mach Freeze Time
RCD	Desired Rate of Climb

2. External Outputs: None

g. Program Entrances

1.	BAL, 15	BIFMCK	Pointer to parameter table in lower halfword of R1 and number of parameters in upper halfword of R1
2.	BAL, 15	CFGCHECK	Pointer in R1 to doubleword which contains bit positions to be checked
3.	BAL, 15	CTO:2	-

h. Exits

1.	B	*15	calling location +1 if all parameters within limits calling location +2 if any parameter out of limits
2.	B	*CFGSAV	calling location +1 if desired configuration calling location +2 if not desired configuration
3.	B	*CTOSAV	

i. Subroutines Called. QINSERT COGNITRONICS Output

j. Memory Requirements

1. Instructions 68
2. Data 27

k. Type of Program Module

1. BIFMCK — Subroutine
2. CFGCHECK — Subroutine
3. CTO:2 — Background

l. Flow Charts. See figure 2-29.

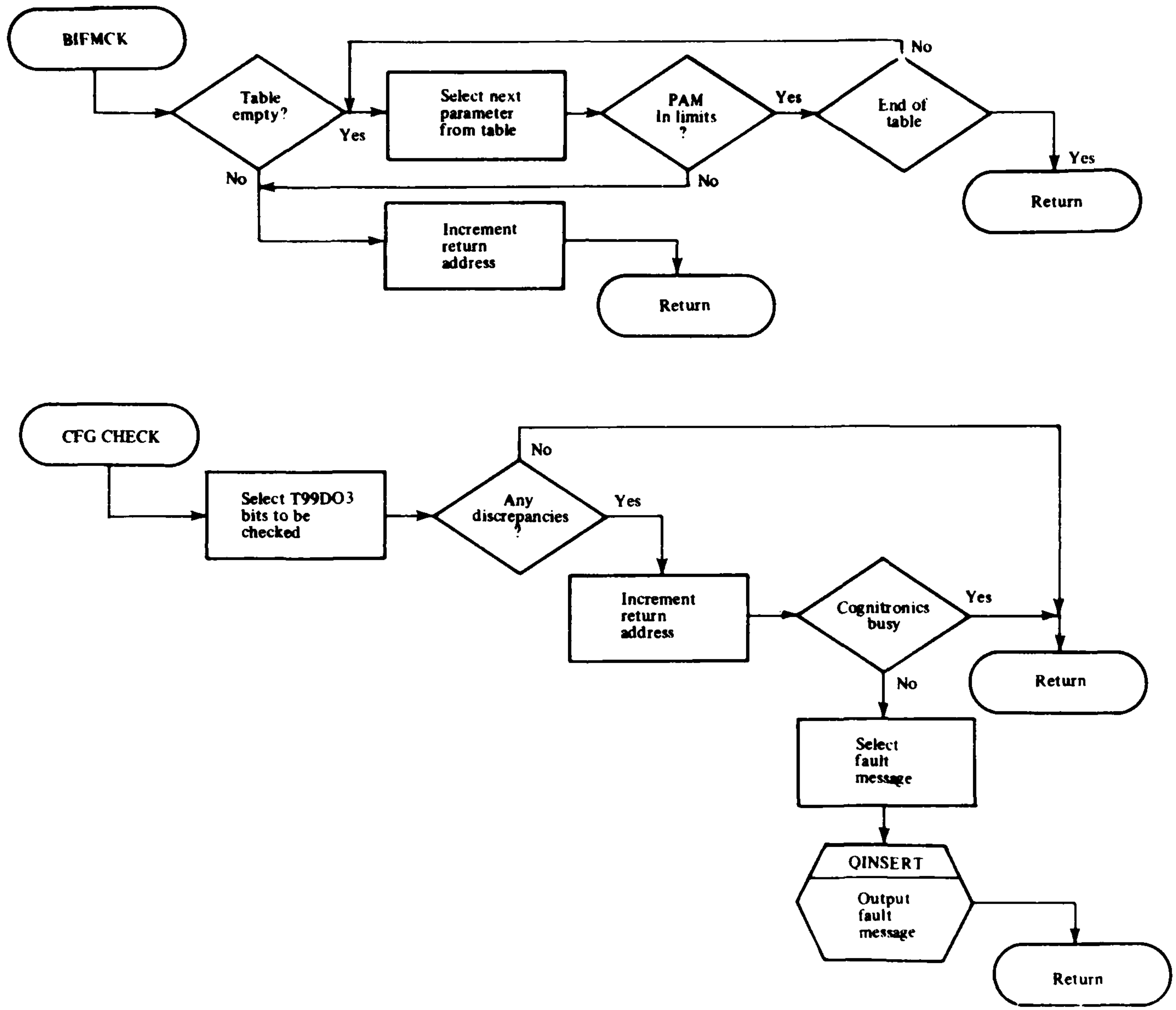


Figure 2-29. Control Aircraft Routines (Sheet 1 of 2)

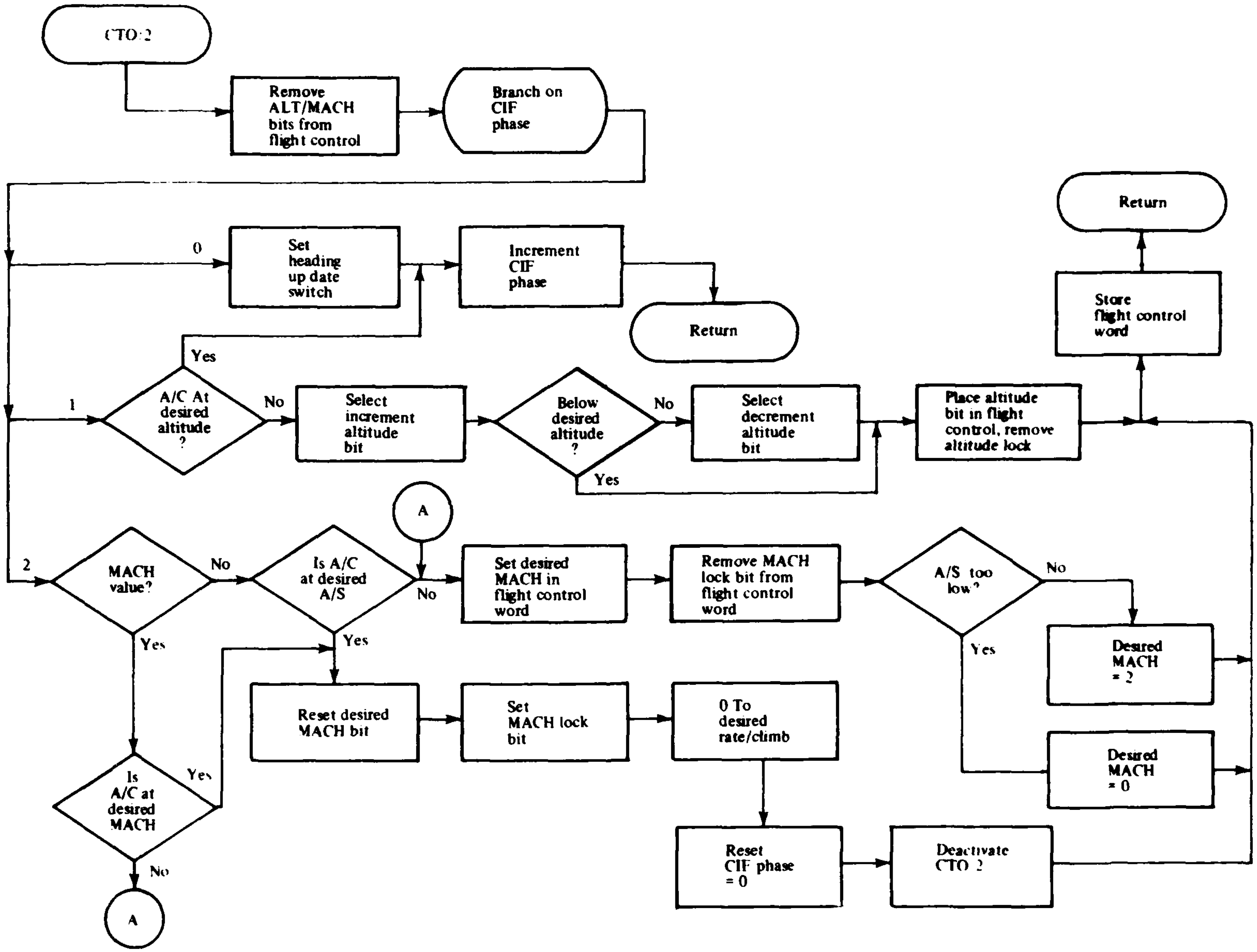


Figure 2-29. Control Aircraft Routines (Sheet 2 of 2)

2.16 NAVIGATION/STRIKE TASK SELECTOR

a. Program Module Name. Navigation/Strike Task Selector (NSTS)

b. Purpose. The purpose of the NSTS module is to select the next task to be run by the student trainee.

c. Requirements. The NSTS module is required to:

1. Select the next task from data accumulated in the student file and task description tables.

2. Select the difficulty factors for the next run or leg.

3. Select and implement the initialization data for various parameters.

4. Initialize the Parameter Description Tables (PDT) for the run or leg selected.

5. Select the COGNITRONICS briefing and instruction messages needed by the trainee prior to start of the run.

6. Initialize all data buffers required by the task.

7. Provide for reinitialization of parameters if the run consists of more than one leg.

8. Initialize the graphics displays.

d. Description. The NSTS subroutine module is called by the Exercise Scheduler module. If the task consists of more than one leg, subsequent legs are initiated by a call to the NSTSR entry point.

NSTS interprets the directives contained in the Task Description Tables (TDT's). It retrieves the next task to be run from the student file and selects the appropriate TDT and difficulty levels (refer to the Task Description Parameter Module, paragraph 2.4.d for a description of the TDT). The program then interprets the TDT and performs the action specified. These actions may include any of all of the following:

1. Selection and initialization of the Parameter Description Tables needed by the Task Performance Monitor (TPM) Module.

2. Selection of the run time and/or leg time.

3. Selection of the specified briefing and initialization messages, via the COGNITRONICS.

4. Selection of the required configuration for the aircraft at the start of the run.

5. Selection of the required system parameter limits which permit the run to start.

6. Selection of the terminating conditions for the run.

7. Selection of the initial assigned values of heading, altitude, and airspeed.

8. The retention of the data required for additional legs for this run, if a multi-leg task.

9. Selection of initial data required for the display buffers.

10. If an attack task, the selection of the required target data; i.e., initial range, bearing, heading, etc.

11. Selection of data table control functions.

Once the data in the Task Description Table has been processed, the program initializes the IDIIOM display, the line printer output buffers, the COGNITRONICS output queue, and the task control parameters.

e. Inputs

1. Internal Inputs:

ATGATE	Attack Gate Penetration Switch
DISBLNK	ASCII Blanks
CEXER	Current Exercise
FCONTFL	Fuel Lock Control
FLEVL	STUDENT FILE – Difficulty Level
ILPTR	Pointer to Line Printer Task Title Output Table
NSTPTR	Pointer to Navigation/Strike Task Table

PSIS	Pseudo-Heading
PLMSG	Pointer to Line Printer Title Message
PMBUFPTR	Pointer to Parameter Buffer
PMTAB	Parameter Monitor Table
ROUTE	Task Number
SFNUM	Student File Number
XINI	IDIIOM X-Axis Time Increment

2. External Inputs: CLOCK F-4 Cycle Time

3. Constants:

AHDGINC	Attack Heading Increment
DPINC	Data Processing Time Increment
D1	1
D180	180
D360	360
MASKL16	Lower 16-Bit Mask
MASKU1	Upper 1-Bit Mask
MASKU16	Upper 16-Bit Mask
NSDLTAB	Table of Difficulty Levels
NSTLLM	Difficulty Level Limits
PDTIME	10 Seconds
PTAB	Table of Legal Parameters Addresses

f. Outputs

1. Internal Outputs:

ABMSG	Attack Briefing Message

ABMSGR	Repeat Attack Briefing Message
ADMGS	Attack Directive Message
AFTIME	Run Time
AIMSG	Attack Instruction Message
ATHDG	Attack Set-up Heading
ASPLIM	Attack Set-up Heading Limits
BKTIME	SDM Module Break Time
BMSG	Briefing Message
CFGSTART	Required T99DO3 Start Configuration
CHKTIME	Check Parameter Start Time
CNTTAB	Parameter Sample Table Count
CPS01	Initial Heading Message Buffer
CTRAD	Attack Turn Radius
CTRADLIM	Attack Turn Radius Limits
DAFTALT	Desired AFT Altitude
DAFTAS	Desired AFT Airspeed
DALT	Initial Altitude
DEHDG	Descent Task Heading
DFNEXT	Difficulty Factor Next Leg
DISPT	Display Timer
DPTIME	Data Processing Time
DRTIME	Desired Run Time
FCONT3	Flight Control Word

FIRESW	Missiles Fired Switch
GPMABORT	Run Abort Switch
IDIFIRST	IDIIOM First Time Switch
ILMSG2T	IFM Line Printer Message
ILMSG4T	IFM Line Printer Message
ILMSG5T	IFM Line Printer Message
ILMSG7T	IFM Line Printer Message
ILMSG8T	IFM Line Printer Message
ITBRG	Initial Target Bearing
ITRNG	Initial Target Range
KILLSW	Target 'Kill' Switch
LCOUNT	Line Printer Line Count
LEGN	Number of Legs in Task
LEGSCORE	Leg Score
LFG	Chord Distance to Final Gate
NOMEN	Nomenclature Values
NSTD1	Display Buffer
NSTD3	Display Buffer
PMSG24	Teletype Message
PSIAFT	Desired AFT Heading
PSISU	Heading Update Switch
RBMSG	Repeat Briefing Message
REPTR	NSTS Re-Entry Pointer
RUNTERM	Run Termination Switch
RUNTIME	Maximum Run Time

SCORE	Run Score
SDMTERM	Attack Completed Switch
SDPHASE	Steering Dot Display Phase
SMTAB	Parameter Sample Table
SPAMCFG	Required Starting Parameter Configuration
STDSW	1st Time STD Switch
TDFR	Target Distance at Time of Firing
TDLO	Target Distance at Time of Lock-on
TDTNEXT	Next Pointer to Task Description Tables
TERMCODE	Termination Code
TIMIST	Timer Switch
TPAMCFG	Required Terminating Parameter Configuration
TSTKNOM	Nominal Time for Attack
TURNDIR	Direction to Turn
XAXIS	IDIIOM X-Axis Time Value

2. External Output: None

g. Program Entrances. BAL, 15 NSTS
BAL, 15 NSTSR (LEG REENTRY)

h. Exits. B *NSTEX

i. Subroutines Called

1. CLRDSP — Clear IDIIOM Display
2. COGADDR — Convert Floating Point to COGNITRONICS Addresses
3. DFIFM — Select Difficulty Factors for IFM

4. DPLEG Data Processing for Leg Score

5. FLOAT Convert Hexadecimal to Floating Point

6. LPOUT:1 Line Printer Output

7. POUT:1 Teletype Output

8. SDISP0 Reset LDS-1 Display

j. Memory Requirements

1. Instructions 353

2. Data 24

k. Type of Program Module. Subroutine

l. Flow Charts. See figure 2-30.

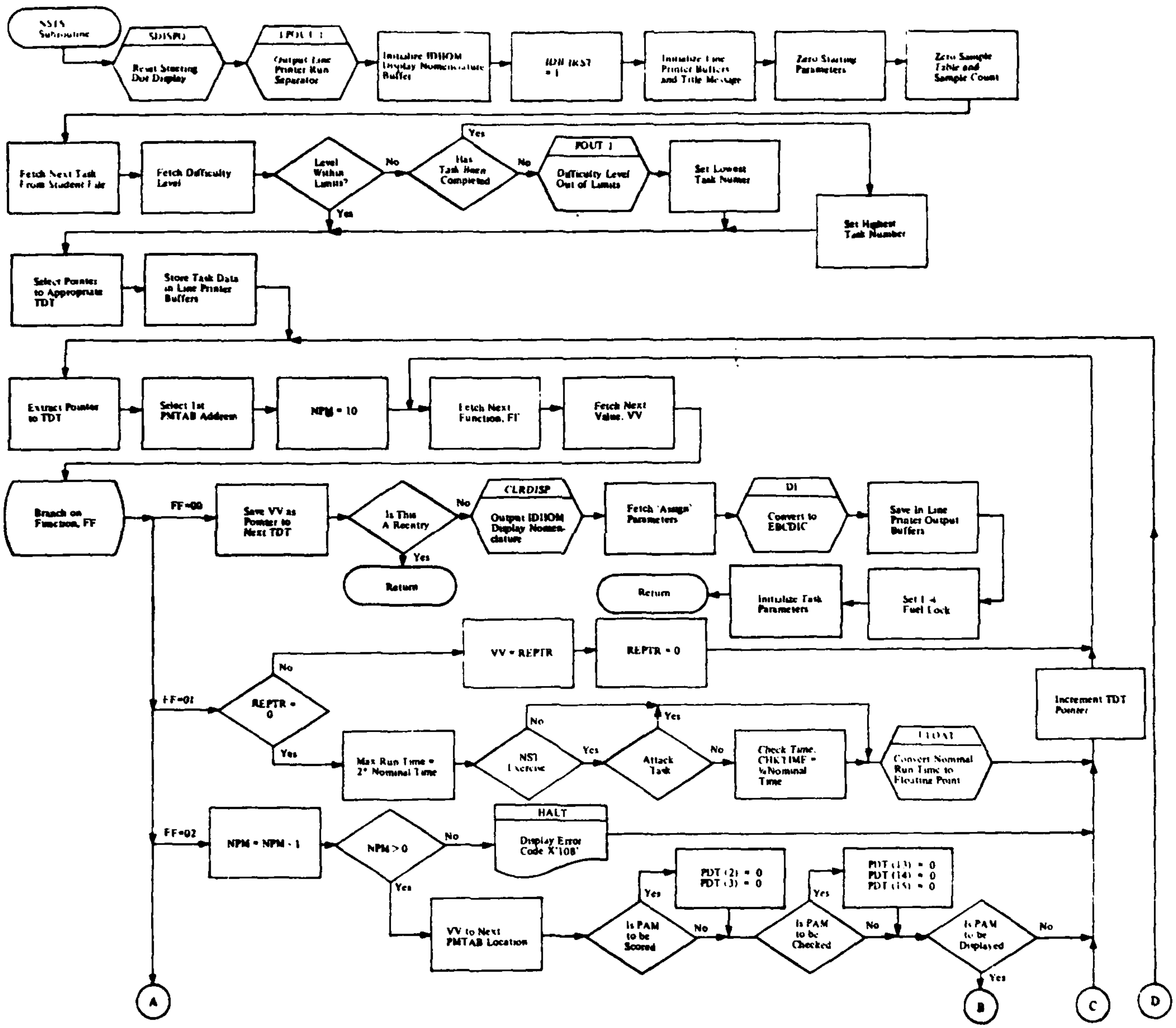


Figure 2-30. Navigation Strike Task Selector (Sheet 1 of 2)

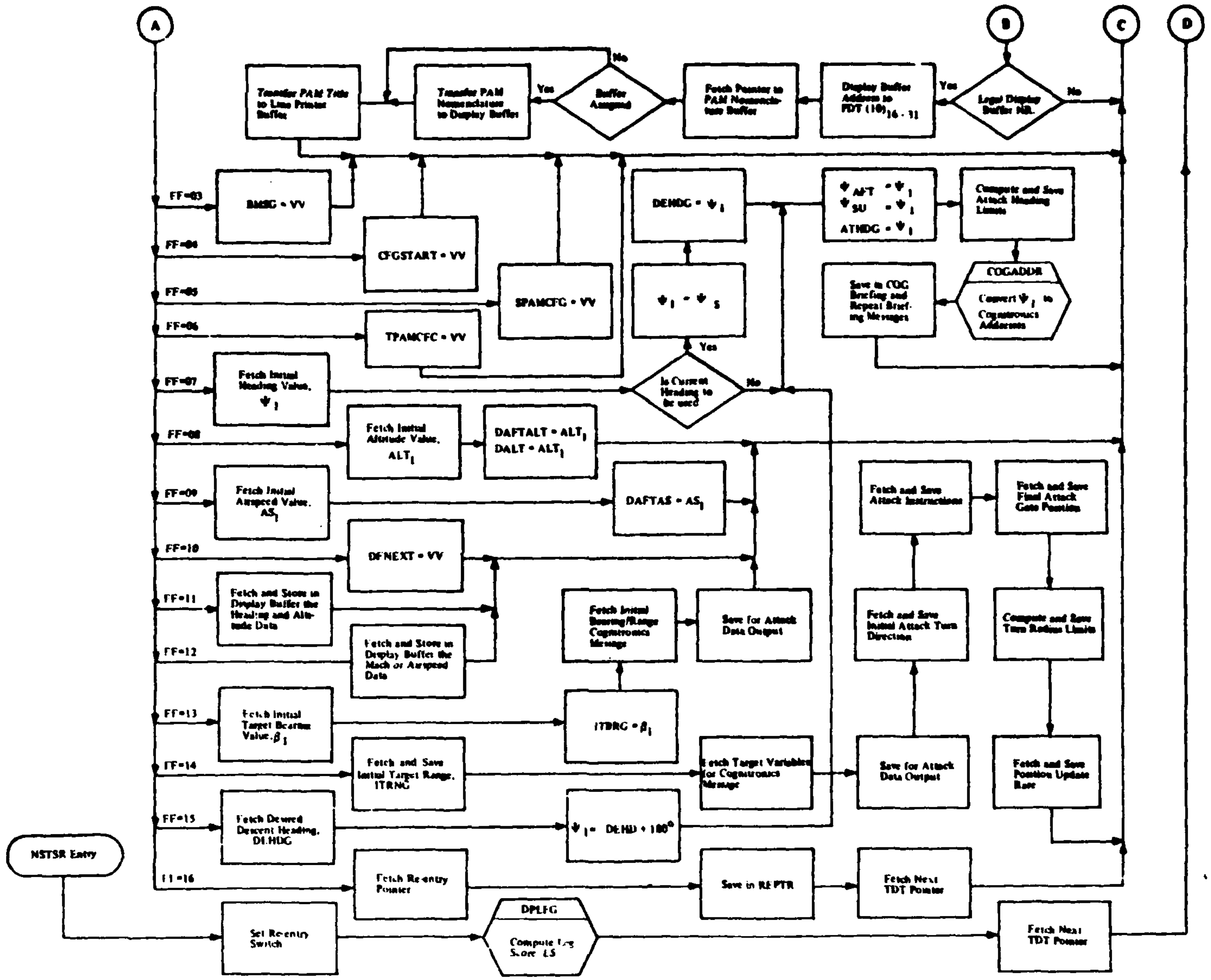


Figure 2-30. Navigation Strike Task Selector (Sheet 2 of 2)

2.17 NAVIGATION/STRIKE PRE-TAKEOFF

a. Program Module Name. Navigation/Strike Pre-Takeoff (NPTO)

b. Purpose. The purpose of the NPTO module is to monitor the aircraft from the time of initialization until airborne.

c. Requirements. The NPTO module is required to:

1. Output the appropriate audio instructions for preflight and takeoff.

2. Trim the aircraft in all three axes for the takeoff.

3. Check for proper takeoff configuration of aircraft.

4. Implement the climb task difficulty factors.

5. Check for simulator in "zero mode."

6. Check for aircraft airborne.

d. Description. The NPTO module is called only if the aircraft is at rest on the ground. *It consists of the following phases*:

1. Phase 0. In this initial phase, the aircraft is trimmed in all three axes for takeoff, and audio instructions are issued to the student to complete the takeoff checklist. The program then advances to Phase 1.

2. Phase 1. In this phase, the student is given 15 seconds to complete the following takeoff checklist items:

   (a) Landing gear handle down.

   (b) Flaps in 1/2 position.

   (c) Engine flame (both engines).

   (d) Speed brake in.

   If, after the 15-second period, the foregoing conditions have not been met, an audio message is sent informing the student of the discrepancy. If all conditions are satisfactory, the program advances to Phase 2.

3. Phase 2. This phase is cycled through until the RESET-TO-ZERO button on the instructor's console is placed in the "out" position.

4. Phase 3. This phase is used to check the following:

   (a) The compass heading has stabilized to that required for takeoff.

   (b) The engines are at minimum idle RPM.

   (c) The aircraft has been trimmed to the proper takeoff. configuration.

   When the foregoing conditions have been met, the climb difficulty factors are implemented, the audio message "cleared for takeoff" is issued, and the program advances to Phase 4.

5. Phase 4. This phase checks to see if the aircraft has become airborne. If not, the engine RPM is checked and, if not at takeoff RPM, the "cleared for takeoff" message is repeated periodically. Once the aircraft is airborne, the Climb Monitor (NSCM) module is activated, and the NPTO module is deactivated.

e. Inputs

1. Internal Inputs:

CMHEAD	Head of COGNITRONICS Message Queue
NATERAF	Table of Next Difficulty Factors
TOPHASE	NPTO Phase
TORPM	Minimum Takeoff RPM
TRIMACT	Trim Subroutine Active Switch

2. External Inputs:

CLOCK	F-4 Cycle Counter
NIL	Engine RPM - Left
NIR	Engine RPM - Right
T99D13	Discrete Input Word 3

T99D01	Discrete Output Word 1
T99D03	Discrete Output Word 3

3. Constants:

ABMASK	On-Ground Bit
NIDLE	Minimum RPM for Idle
PFDES	Desired Preflight Bit Configuration
PFBIT	Preflight Bits to be Checked
PFDT	Preflight Check Time Constant
PFMASK	Bits Checked for Preflight

f. Outputs

1. Internal Outputs:

ATERAF	Table of Current Difficulty Factors
CLPHASE	Climb Task Module Phase
PSIAFT	Pseudo-Heading
PSIUP	Heading Update Switch

2. External Outputs: None

g. Program Entrances. BAL, 15 NPTO:1

h. Exits. B *NPTORET

i. Subroutines Called

1. NSDF Select Difficulty Factors
2. QINSERT COGNITRONICS Output
3. TRIM Trim Aircraft

j. Memory Requirements

1. Instructions 90

2. Data 1

k. Type of Program Module. Background

l. Flow Charts. See figure 2-31.

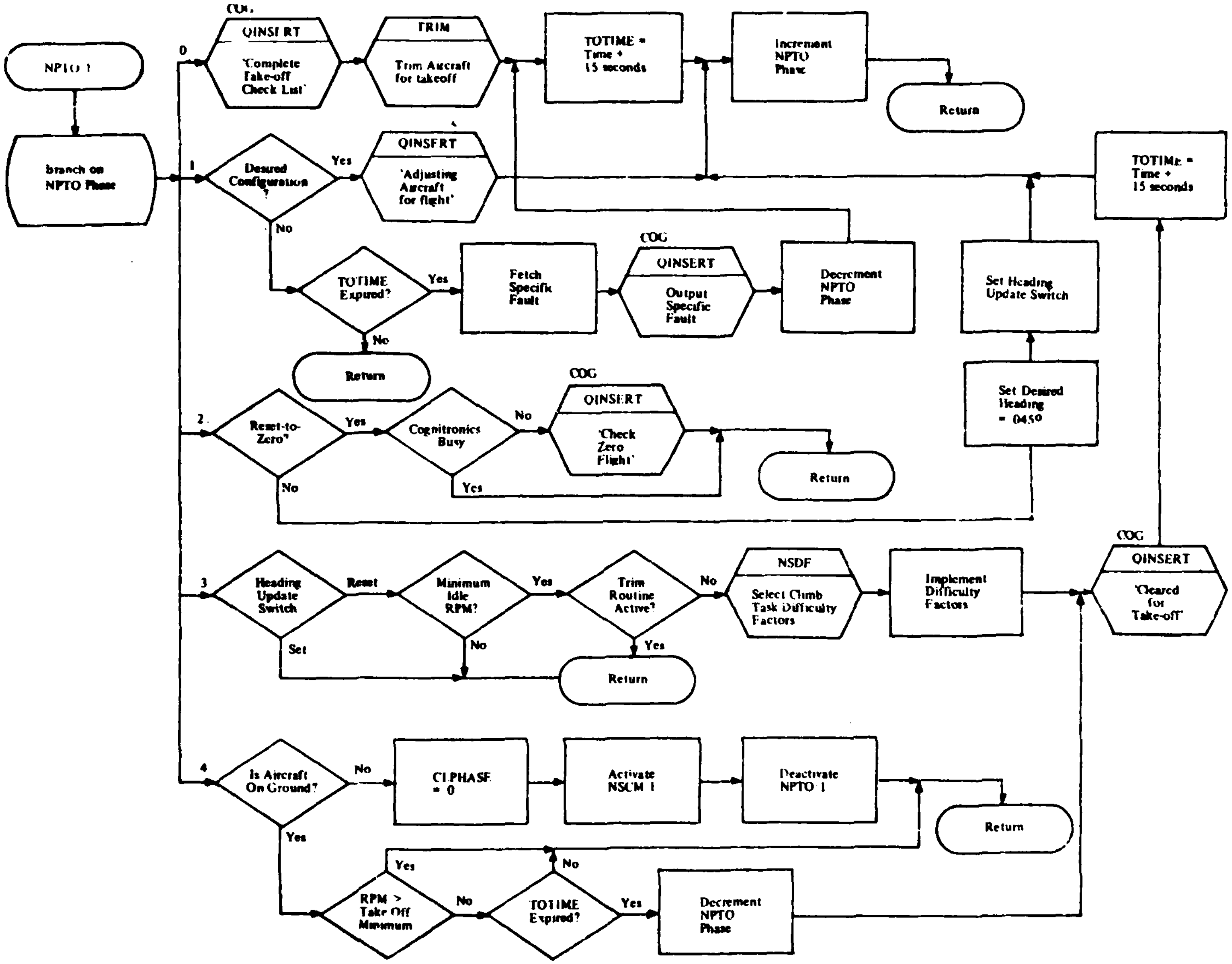


Figure 2-31. Navigation/Strike Pre-Take Off Routine

2.18 NAVIGATION/STRIKE CLIMB MONITOR

a. Program Module Name. Navigation/Strike Climb Monitor (NSCM)

b. Purpose. The purpose of the NSCM module is to monitor the student's actions during the specified climb task.

c. Requirements. The NSCM module is required to:

1. Output the climb entry parameters to the line printer.

2. Output each set of climb leg audio instructions to the student.

3. Determine if the climb leg task has been successfully completed or if it is to be aborted.

d. Description. The NSCM module consists of the following phases:

1. Phase 0. Upon entry, the selected aircraft parameters (heading, airspeed, etc.) sampled at the start of the climb task are output to the line printer. The audio instructions for the selected climb are then output via the COGNITRONICS Speechmaker and the program advances to Phase 1.

2. Phase 1. This phase computes and saves the maximum time allowed for the climb leg and activates the Timer (TIMER:1) and Task Performance Monitor (TPM:1) modules.

3. Phase 2. This phase is looped through until it is determined if the climb leg is to be either aborted or has been successfully completed. If the leg is to be aborted, the program is advanced to Phase 3. If the leg was successfully completed, a test is made to see if another leg is to follow. If not, Phase 3 is set. If another leg is to follow, the Navigation/Strike Task Selector (NSTS) module is called to select the parameters needed by the next climb leg and the Phase 2 monitoring is continued for the next leg.

4. Phase 3. This is the climb termination phase. Audio instructions are issued to the student to inform him of the termination of the climb task, the Task Terminator (TSTR:1) module is activated, and the Timer (TIMER:1), Task Performance Moniter (TPM:1), and Climb Monitor (NSCM:1) modules are deactivated.

Table 2-1 in Paragraph 2-4 (Navigation/Strike System Parameters) lists the climb tasks available to the program.

e. Inputs

1. Internal Inputs:

BMSG	Briefing Message
CHKTIME	Check Parameter Start Time
CLPHASE	Climb Monitor Phase
CMHEAD	Head of COGNITRONICS Message Queue
GPMABORT	Run Abort Switch
RUNTERM	Run Termination Switch
TDTNEXT	Next Pointer to Task Description Tables

2. External Inputs:

CLOCK	F-4 Cycle Counter

3. Constants:

CMSGCC	Climb Termination Audio Message

f. Outputs

1. Internal Outputs:

RSTIME	Run Start Time
RUNTIME	Maximum Run Time

2. External Outputs: None

g. Program Entrances. BAL, 15 NSCM:1

h. Exits. B *NSCMRET

i. Subroutines Called

1.	CFGCHECK	Check Aircraft Configuration
2.	ENTROUT	Output Entry Parameters
3.	NSTSR	Navigation/Strike Task Selector (Re-entry)
4.	QINSERT	COGNITRONICS Output

j. Memory Requirements

1.	Instructions	54
2.	Data	4

k. Type of Program Module. Background

l. Flow Charts. See figure 2-32.

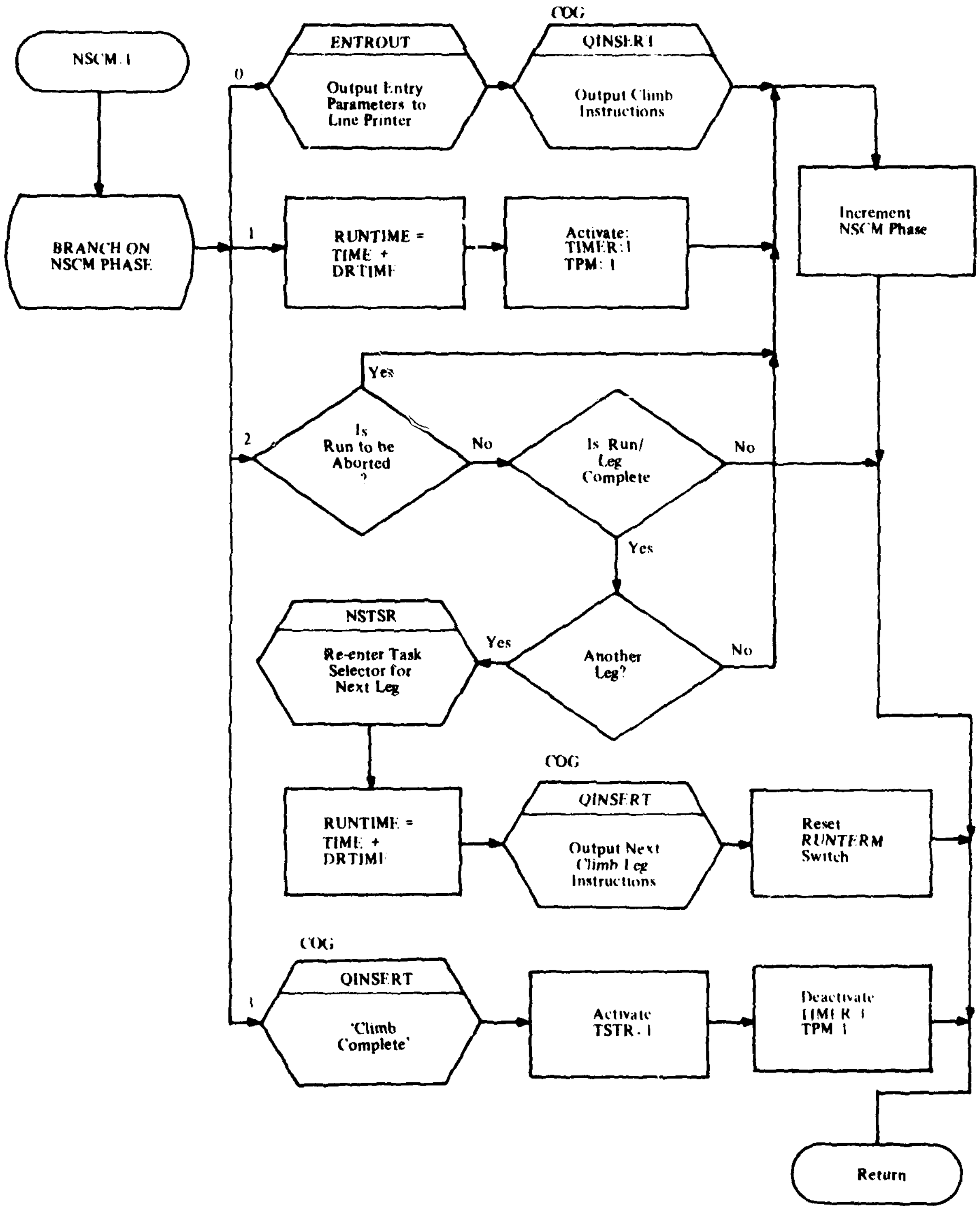


Figure 2-32. Navigation/Strike Climb Monitor

2.19 NAVIGATION/STRIKE ATTACK MONITOR

a. Program Module Name. Navigation/Strike Attack Monitor (NSAT)

b. Purpose. The purpose of the NSAT module is to monitor the student's actions during the specified attack task.

c. Requirements. The NSAT module is required to:

1. Output the attack briefing, data, and command audio messages.

2. Implement the specified difficulty factors.

3. Check to ensure the aircraft is in the specified starting configuration to begin the attack.

4. Output the attack entry parameters to the line printer.

5. Compute the target look-angle values.

6. Monitor the target's position relative to the interceptor, compute the vectors necessary to intercept the target, and output the resultant audio commands to the student.

7. Activate the final attack phase (SDM: 1 module) if the student successfully penetrates the final attack gate.

8. Abort the run if the student fails to maintain the attack vector envelope.

d. Description. The NSAT module consists of the following phases:

1. Phase 0. In this phase, the difficulty factors are implemented and the attack briefing message is sent via the COGNITRONICS audio output. The briefing message issues the initial interceptor attack heading and airspeed plus the target composition, altitude, heading, and airspeed.

2. Phase 1. This phase waits until the interceptor is at the correct heading, altitude, and airspeed. If, within 90 seconds, these conditions have not been met, the attack briefing message is repeated every 15 seconds until the correct altitude, heading, and airspeed are attained. At this point, the attack data message is transmitted. The attack data message supplies the student with the target's initial range and bearing and informs him to take radar control for the final attack.

3. Phase 2. This phase waits until the transmission of the attack data message has been completed, outputs the entry parameters to the line printer, and outputs the audio attack instructions to the student. The attack instructions consist of the initial direction and degree of bank angle required to intercept the target.

4. Phase 3. In this phase, the interceptor is monitored for execution of the attack instructions. Once the bank angle exceeds 20° in the proper direction, the program begins to monitor the student's actions and issues the correction commands necessary to intercept the target.

5. Phase 4. This phase monitors and controls the interceptor's actions. The target look-angles are computed and saved by the CLA subroutine. These look angles are used by the Strike Target Dynamics (STD) module along with target range and bearing parameters to determine if the interceptor has penetrated the final attack gate and 'lock-on' has occurred. The Attack Monitor (ATM) subroutine is executed in this phase. This subroutine controls the attack display (IDIAT subroutine), monitors the interceptor's position relative to the nominal path required to intercept the target, and outputs the appropriate audio commands required to keep the interceptor within the nominal path envelope.

   Every 5 seconds, the interceptor's bank angle is sampled, and the turn rate is computed using the following equation:

$$R_{IS} = C_1 \phi_s^{5} + C_2 \phi_s^{3} + C_3 \phi_s$$

   where:

   ϕ_s = present bank angle

   $C_1 = 0.45 \times 10^{-8}$

   $C_2 = -1.284 \times 10^{-5}$

   $C_3 = 4.791 \times 10^{-2}$

   From the turn rate, the average heading, ψ_{PD}, over the time period, t, is computed

$$\psi_{PD} = 1/2\ (R_{IS})/(t)$$

The predicted position with respect to the nominal path can then be computed (See Figure 2-33) as shown in the following relationships:

$$\Delta X = (V_G)(t)(\cos \psi_{PD})$$

$$\Delta Y = (V_G)(t)(\sin \psi_{PD})$$

where V_G = airspeed

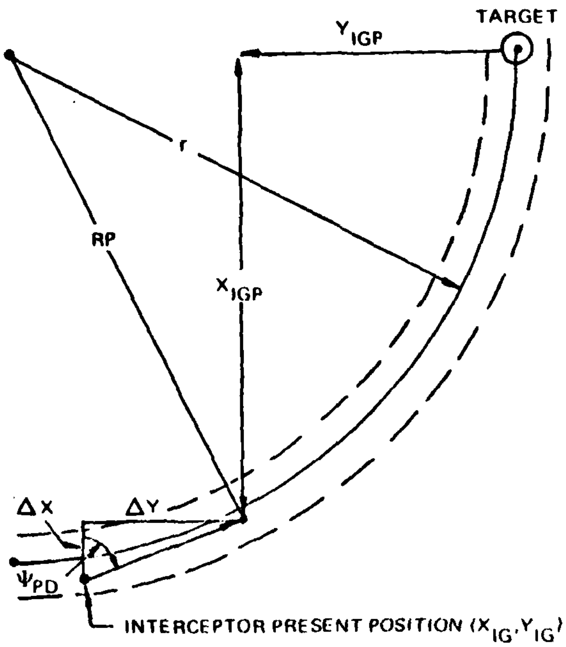


Figure 2-33. Nominal Path Attack Geometry

$$X_{IGP} = X_{IG} + \Delta X$$

$$Y_{IGP} = Y_{IG} + \Delta Y$$

The updated turn radius, RP, can then be computed:

$$RP = \sqrt{(X_{IGP})^2 + (r - (Y_{IGP}))^2}$$

If RP is within the 'Hold Bank'envelope, the command "Hold Bank" is issued. If RP is less than the 'Hold Bank' lower envelope limit, the command "Harder Bank" is given. If RP is greater than the "Hold Bank" upper envelope limit, the command "Ease Bank" is issued. If RP is so large or small that it is impossible for the interceptor to return to the nominal path, the run is aborted. In addition, a maximum time limit for each attack task is computed and, if this limit is exceeded, the run is aborted. If the final attack gate is penetrated (determined by the STD module), the final attack mode is entered (Phase 5). The final attack mode simulates radar lock-on of the target by displaying a simulated radar picture on the LDS-1 display screen mounted in the cockpit. Figure 2-34 illustrates simulated radar display in the final attack mode. Generation of the display is accomplished in the Steering Dot Monitor (SDM) module which is actuated whenever the final attack gate is penetrated.

If final attack gate penetration is achieved, the gate penetration data is output to the line printer, the leg score is computed and saved, and the program is advanced to Phase 5. If the run is aborted, the leg score is computed and Phase 7 is set.

6. Phase 5. This phase is entered whenever the final attack gate is penetrated. Look-angles are computed to determine if radar lock-on has been lost. If so, Phase 7 is set. If not, the SDM program is constantly monitored for successful completion of the attack, at which time Phase 6 is set.

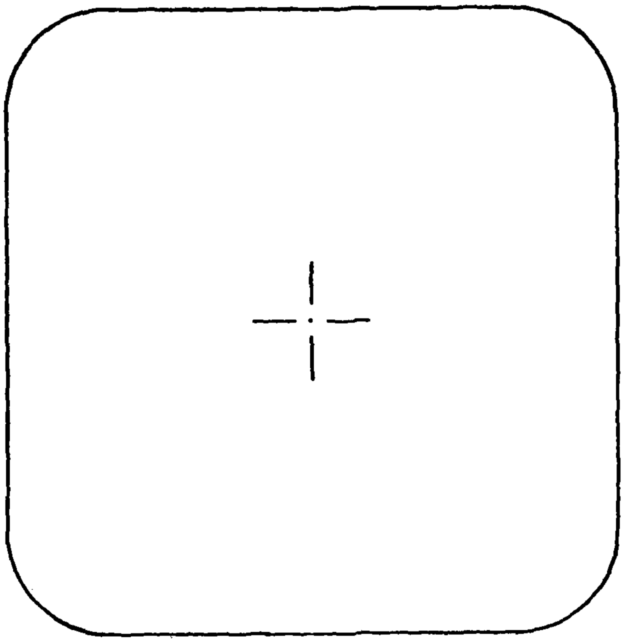
7. Phase 6. If a successful attack was completed, this phase is entered. The audio command "Attack completed" and "Break" instructions are given to the student. The TIMER:1, STD:1, TPM:1, and NSST:1 modules are deactivated and the Task Terminator (TSTR:1) module is activated.

8. Phase 7. This phase is entered whenever the attack task is to be aborted. The audio command "Break off attack" is given to the student and the "Lost lock-on" data is output to the line printer. The TIMER:1, STD:1, TPM:1, and NSST:1 modules are deactivated and the Task Terminator (TSTR:1) module is activated.

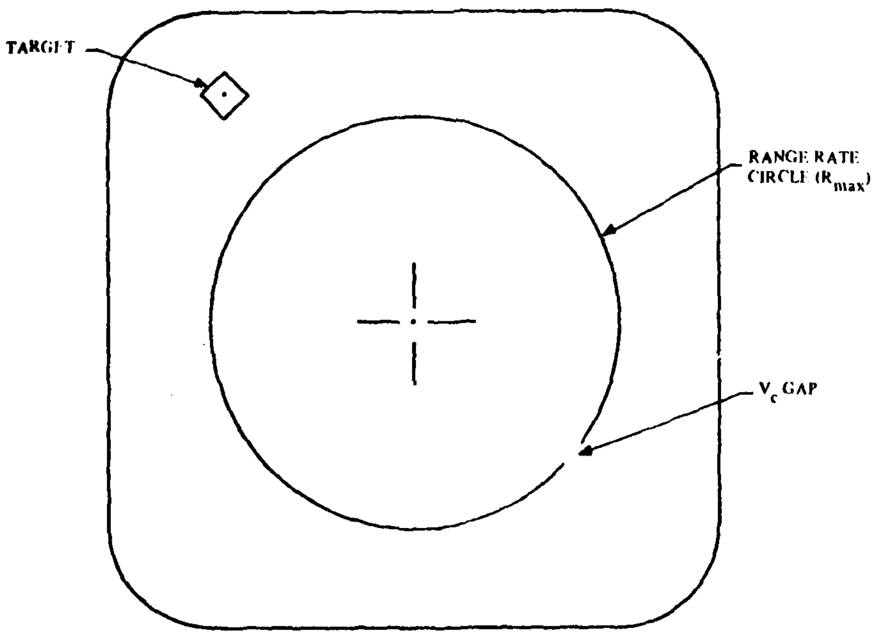
e. Inputs

1. Internal Inputs:

ATGATE	Attack Gate Penetration Switch
CMHEAD	Head of COGNITRONICS Message Queue
COSPD	Cosine ψ_{PD}
CTRAD	Attack Turn Radius
CTRADLIM	Attack Turn Radius Limits

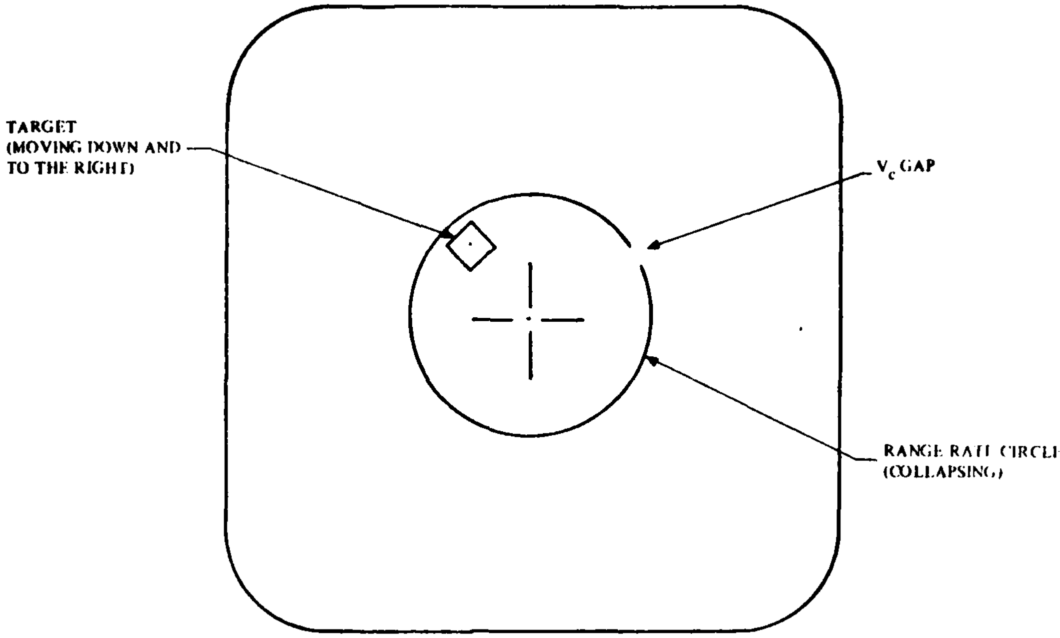


BASIC SCOPE
(PRIOR TO LOCK-ON)

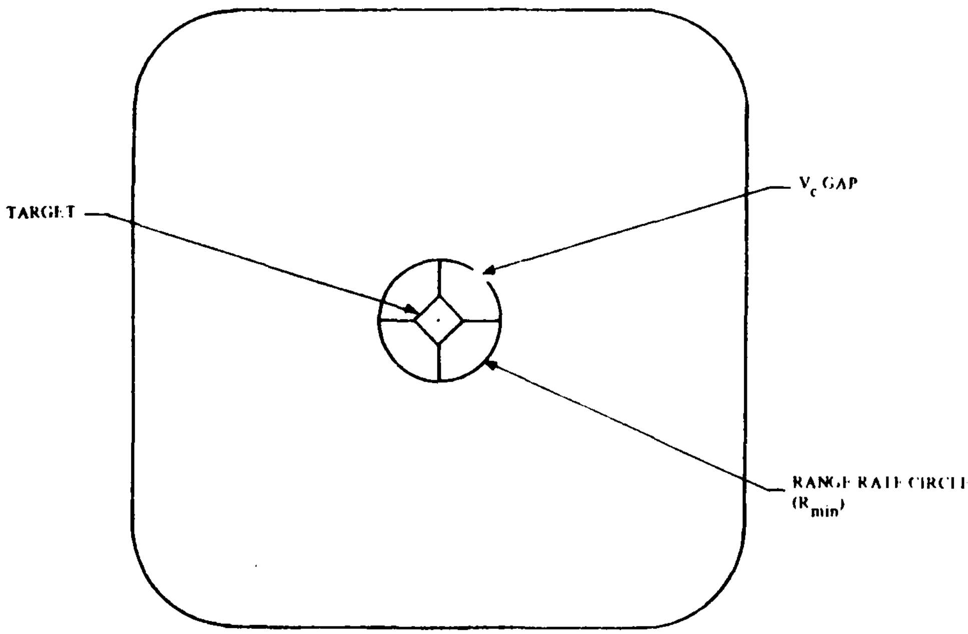


LOCK-ON
(STEER UP AND PORT TO CENTER TARGET)

Figure 2-34. Cockpit Simulated Radar Display (Sheet 1 of 3)

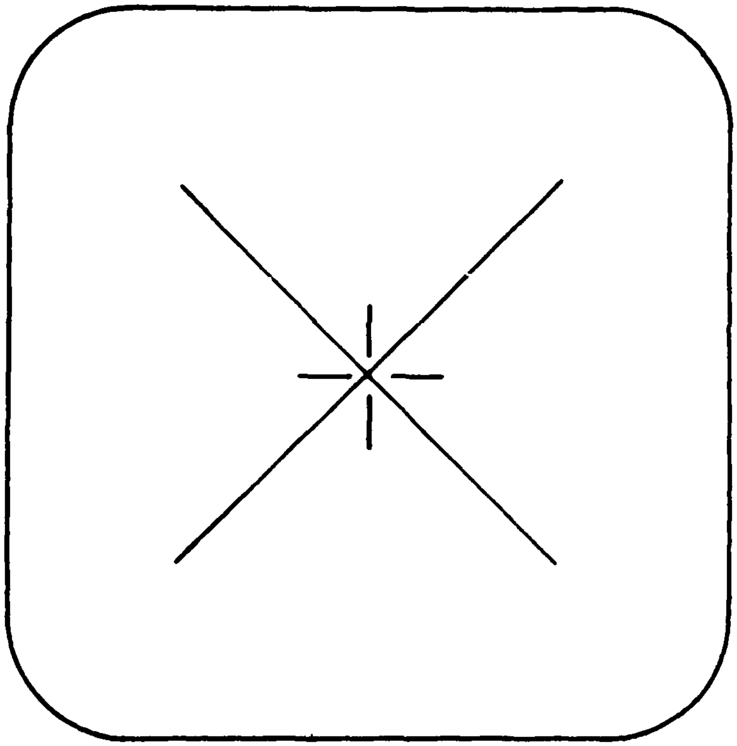


CORRECTION APPLIED
(TARGET MOVING TO CENTER)



IDEAL FINAL ATTACK DISPLAY

Figure 2-34. Cockpit Simulated Radar Display (Sheet 2 of 3)



ATTACK COMPLETE
(BREAK AWAY)

Figure 2-34. Cockpit Simulated Radar Display (Sheet 3 of 3)

NATERAF	Table of Next Difficulty Factors
PDTIME	Predicted Position Time Increment
PFDT	Repeat Briefing Message Time Increment
PHIMIN	Minimum Bank Angle
PHIS	Bank Angle
PSIS	Pseudo-Heading
RUNTIME	Maximum Run Time
SDMTERM	Attack Completed Switch
SINPD	Sine ψ_{PD}
SPAMCFG	Required Starting Parameter Configuration
STDSW	1st Time (STD:1) Module Switch
STINC	Initial Repeat Briefing Message Time Increment
STPHASE	NSST Phase Number
TSTK	Strike Run Time
TSTKNOM	Nominal Time for Attack Task
TURNDIR	Direction to Turn
XI	Distance Behind Target (this cycle)
XIG	Distance Behind Gate Position (this cycle)
YI	Distance Right of Target (this cycle)
YIG	Distance Right of Gate Position (this cycle)
ZI	Distance Below Target (this cycle)

2. External Inputs:

CLOCK	F-4 Cycle Counter
DSATRIM	Actual Aileron Trim Value
VG	Interceptor Velocity
XT	Actual Longitudinal Turn Value

3. Constants:

ADMSG	COGNITRONICS Attack Data Message
AIMSG	COGNITRONICS Attack Instructions Message
ATINC	Attack Time Increment

BMSG	COGNITRONICS Briefing Message
BKMSG01	COGNITRONICS "Port" Message
BKMSG02	COGNITRONICS "Starboard" Message
CMSGAC	COGNITRONICS Attack Completed Message
CMSGBRK	COGNITRONICS Break Off Attack Message
CMSGEBK	COGNITRONICS Ease Bank Message
CMSGHBK	COGNITRONICS Hold Bank Message
CMSGIBK	COGNITRONICS Increase Bank Message
D180	180 Degrees
D360	360 Degrees
KNMFT	Nautical Miles to Feet Conversion Factor
RBMSG	COGNITRONICS Repeat Briefing Message

f. Outputs

1. Internal Outputs

AFTIME	Final Attack Mode Run Time
ATERAF	Table of Current Difficulty Factors
ATTIME	Initial Attack Mode Maximum Run Time
CHKTIME	Check Parameters Starting Time
COSLA	Cosine of Horizontal Look-Angle
CSCSW	Compute Sin/Cos Switch
GPMABORT	Abort Run Switch
LOOKSW	Look-Angle Data Ready Switch
PSIPD	Average Predicted Heading
RSTIME	Run Start Time
SINLA	Sine of Horizontal Look-Angle
SINVA	Sine of Vertical Look-Angle
STIME	Strike Time
TERMCODE	Run Termination Code

2. External Outputs: None

g. Program Entrances. BAL, 15 NSST;1

h. Exits. B *NSSTRET

i. Subroutines Called.

AFDOUT	Attack Firing/Lost Lock-On Data Output
ATM	Attack Monitor
BIFMCK	Check Starting Parameters
CHKCON	Check Wheels/Flaps
CLA	Compute Look-Angles
DPLEG	Compute Leg Score
ENTROUT	Output Entry Parameters
GDOUT	Gate Penetration Data Output
IDIAT	Initial Attack Display
NSDF	Implement Difficulty Factors
QINSERT	COGNITRONICS Output
SQRTO	Square Root
TRIM	Trim Aircraft

j. Memory Requirements

1. Instructions 266
2. Data 17

k. Type of Program Module. Background

l. Flow Charts. See Figures 2-35 through 2-37.

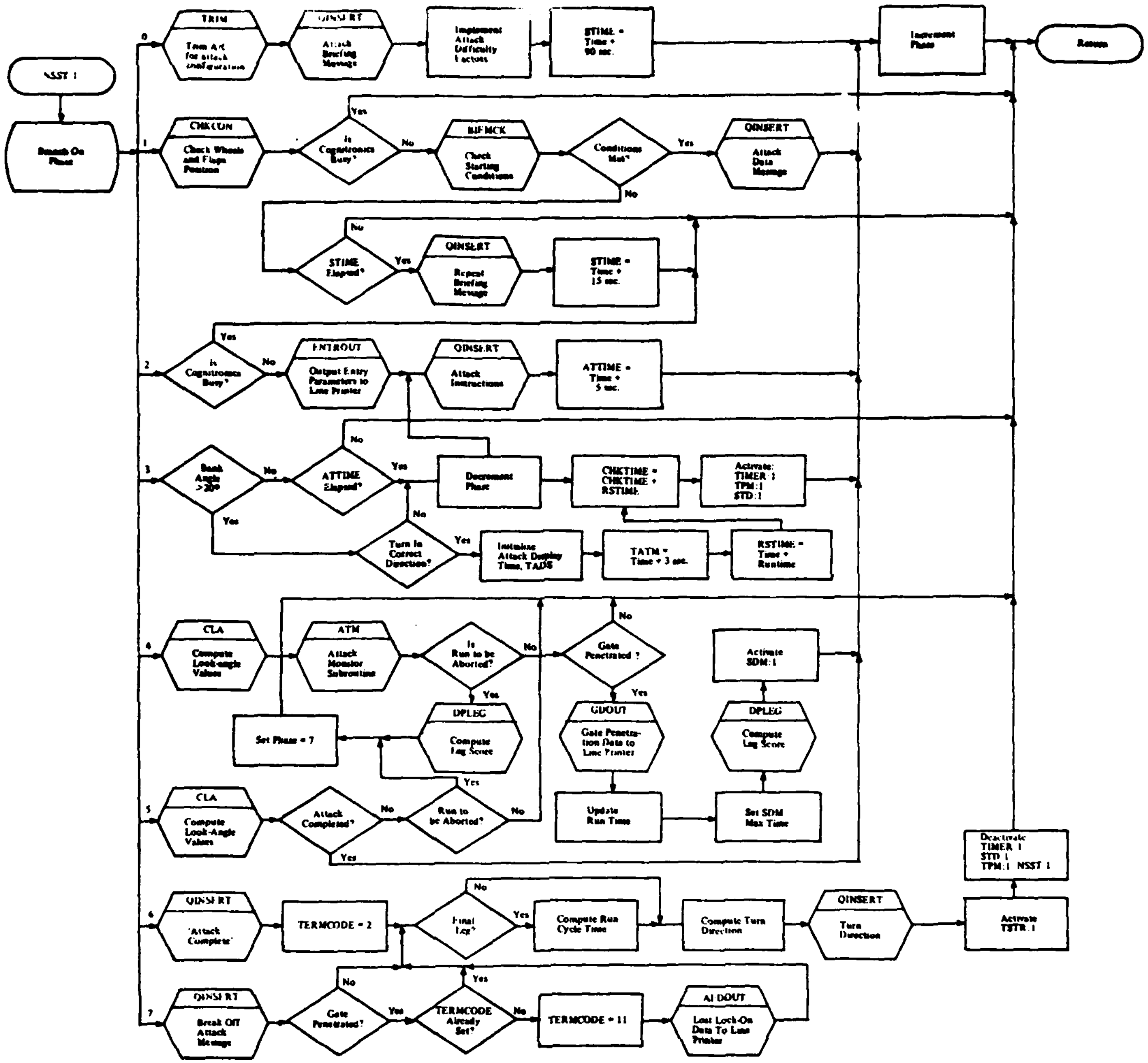


Figure 2-35. Navigation/Strike Monitor

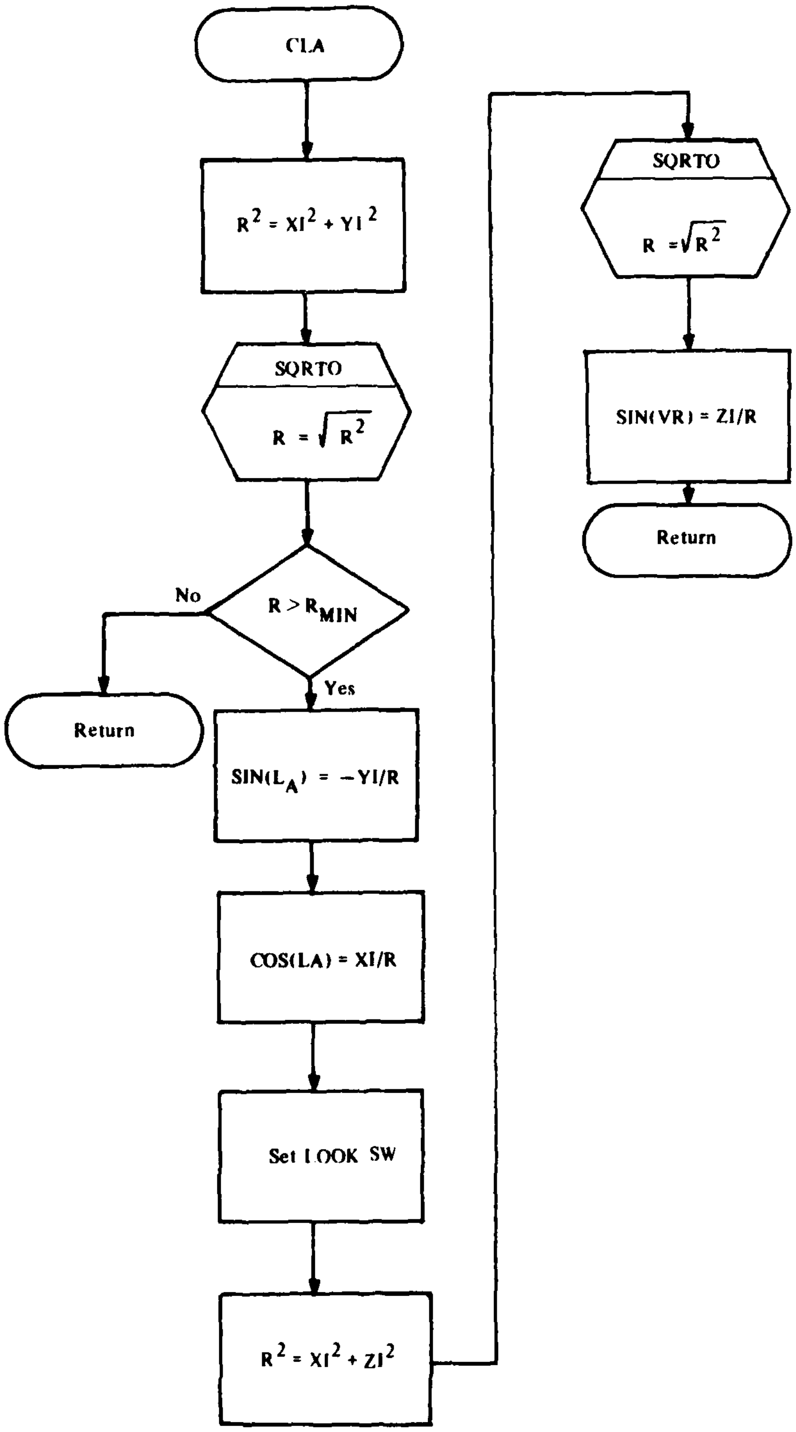


Figure 2-36. Compute Look Angles Subroutine

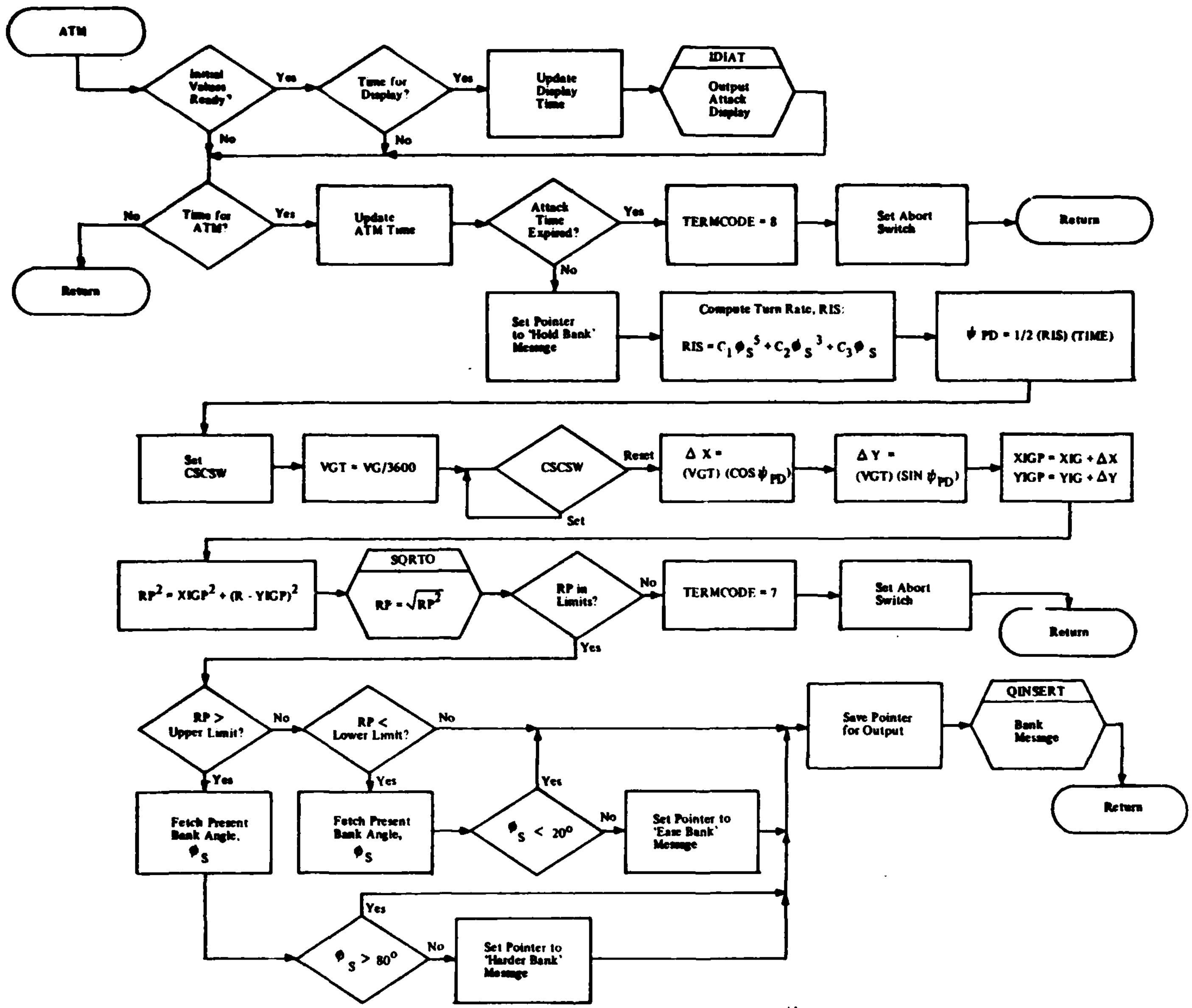


Figure 2-37. Attack Monitor Subroutine

2.20 NAVIGATION/STRIKE DESCENT MONITOR

a. Program Module Name. Navigation/Strike Descent Monitor (NSDE)

b. Purpose. The purpose of the NSDE module is to monitor the student's actions during the specified descent task.

c. Requirements. The NSDE module is required to:

1. Select and implement the required difficulty factors.

2. Output the descent entry parameters to the line printer.

3. Output each set of descent leg audio instructions to the student.

4. Determine if the descent leg task has been successfully completed or if it is to be aborted.

d. Description. The NSDE module consists of the following phases:

1. Phase 0. In this phase, the appropriate difficulty factors are selected by the NSDE subroutine and then implemented. The entry parameters are output to the line printer and the next leg's audio instructions are output to the student.

2. Phase 1. This phase computes and saves the maximum time allowed for the descent leg and then activates the Timer (TIMER:1) and Task Performance Monitor (TPM:1) modules.

3. Phase 2. This phase is used as a check to see if the student has responded to the throttle settings specified. Actual engine RPM's are compared against the minimum required for the descent task. At the end of a given time period, if the actual RPM's are greater than the minimum, the audio briefing message is repeated. Whenever the actual RPM's fall below the minimum, the program is advanced to Phase 3.

4. Phase 3. This phase is looped through until it is determined if the descent leg is to be either aborted or has been successfully completed. If the leg is to be aborted, the program is advanced to Phase 4. If the leg was successfully completed, a test is made to see if another leg is to follow. If not, Phase 3 is set. If another leg is to follow, the Task Selector (NSTS) module is called to select the parameters needed by the next descent leg and Phase 3 monitoring is continued for the next leg.

5. Phase 4. This is the descent termination phase. Audio instructions are issued to the student to inform him of the termination of the descent task, the Task Terminator (TSTR:1) module is activated, and the Timer (TIMER:1), Task Performance Monitor (TPM:1), and Descent Monitor (NSDE:1) modules are deactivated.

Table 2-3 in Paragraph 2.4 illustrates the descent tasks available to the program.

e. Inputs

1. Internal Inputs:

BMSG	Briefing Message
CHKTIME	Check Parameter Start Time
DEPHASE	Descent Monitor Phase
GPMABORT	Run Abort Switch
NATERAF	Table of Next Difficulty Factors
RUNTERM	Run Termination Switch
TDTNEXT	Next Pointer to Task Description Table

2. External Inputs:

CLOCK	F-4 Cycle Counter
NIL	Engine RPM (left)
NIR	Engine RPM (right)

3. Constants:

CMSGDC	Descent Termination Audio Message
RPM80	Minimum Descent RPM

f. Outputs

1. Internal Outputs:

ATERAF	Table of Current Difficulty Factors
RSTIME	Run Start Time
RUNTIME	Maximum Run Time

2. External Outputs: None

g. Program Entrances. BAL, 15 NSDE:1

h. Exits. B *NSDESAV

i. Subroutines Called

1.	CHKCON	Check Wheels/Flaps Up
2.	ENTROUT	Output Entry Parameters
3.	NSDF	Select Difficulty Factors
4.	NSTSR	Navigation/Strike Task Selector (Re-entry)
5.	QINSERT	COGNITRONICS Output

j. Memory Requirements

1. Instructions 66
2. Data 2

k. Type of Program Module. Background

l. Flow Charts. See figure 2-38.

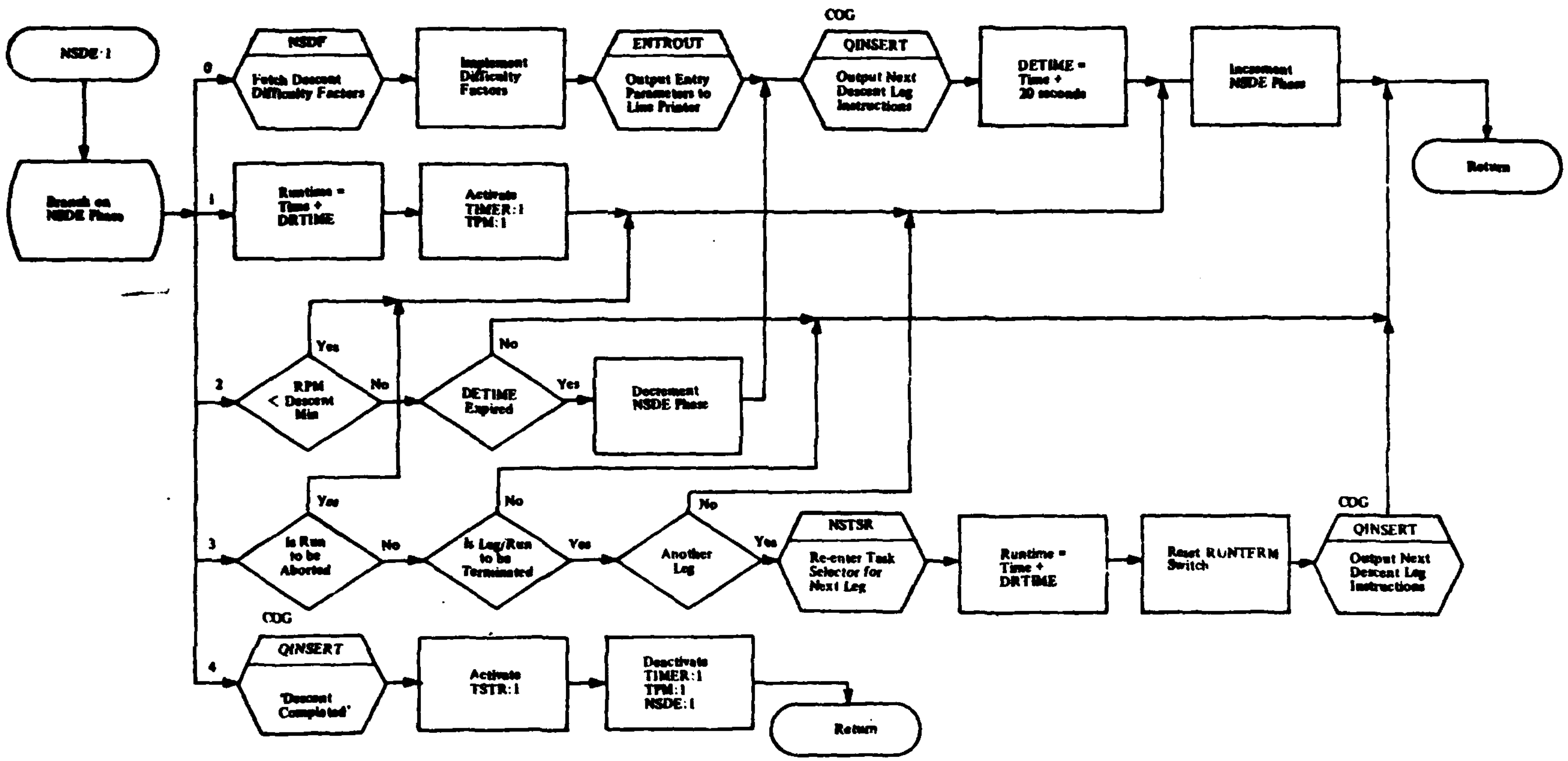


Figure 2-38. Navigation/Strike Descent Monitor

2.21 TASK PERFORMANCE MONITOR

a. Program Module Name. Task Performance Monitor (TPM)

b. Purpose. The purpose of the TPM module is to monitor specified system and aircraft parameters for the selected task and to issue appropriate correctional instructions to the trainee, if necessary.

c. Requirements. The TPM module is required to:

1. Monitor the aircraft and system parameters specified in the Parameter Table (PMTAB).

2. Issue the appropriate COGNITRONICS messages to the trainee if the aircraft is detected outside of the categorized limits specified in the Parameter Description Tables.

3. Terminate the run if any of the following conditions are met:

   (a) Run is started outside of specified parameter limits.

   (b) Expiration of run time.

   (c) Successful run termination conditions are met.

   (d) Aircraft is detected outside of any outer limit specified in the PDT.

4. At specified time intervals (500 ms), sample and save the parameter or parameter error values designated by the PDT.

5. Maintain a history of the limit categories for those parameters specified in the PDT.

6. Output to the display the error displacement for those parameters specified in the PDT.

7. Output to the line printer every 15 seconds the average errors for the specified parameters.

d. Description. The TPM routine monitors the performance of the aircraft via the data contained in the Parameter Description Tables (PDT) selected for the task. Pointers to the selected PDT are contained in the Parameter Monitor Table (PMTAB) for the run in progress, and are set up in the Navigation/Strike Task Selector (NSTS) module. Figure 2-39 illustrates the PMTAB setup. This technique permits the selection of any combination of PDT for any specified task. The PDT sequence and selection are prescribed in the Task Description Tables.

Depending upon the actions specified, any or all of the following functions are performed by TPM:

1. CHECK. If checking is specified, the parameter error is compared to preset limits. If the error falls outside of the outer limits, the run is aborted. If the error is within the outer limits but outside of the middle limits, the appropriate instruction message is selected and saved. Instruction messages inform the trainee to take some type of positive action to bring the aircraft back within the inner envelope. If the error falls within the middle limits but outside the inner envelope, a check message is selected and saved. Check messages inform the trainee to check the specified parameter (heading, altitude, etc.) and take whatever action deemed necessary to bring the aircraft back within the inner envelope. If the error is within the inner envelope, no message is selected.

   In addition, if the parameter is to be checked, each time the parameter is sampled a record is kept as to the number of times it was observed in each of the above categories (inner envelope, middle limits, outer limits) and this data is output every 15 seconds to the line printer.

2. MONITOR ONLY. If monitor only is specified by the PDT, a check is made to see if the error is within the outer limits. If not, the run is aborted.

3. SAMPLE. If the parameter is to be sampled, the average parameter error is accumulated each 500 ms over a 15-second interval and then output to the line printer.

4. SCORE. If scoring is indicated, the absolute error is normalized, squared, and summed over the run as follows:

   $$SUM = \sum P_V (P_C - P_A)^2$$

   where:

   P_C = Parameter, nominal value

   P_A - Parameter, actual value

   P_V = Normalization factor

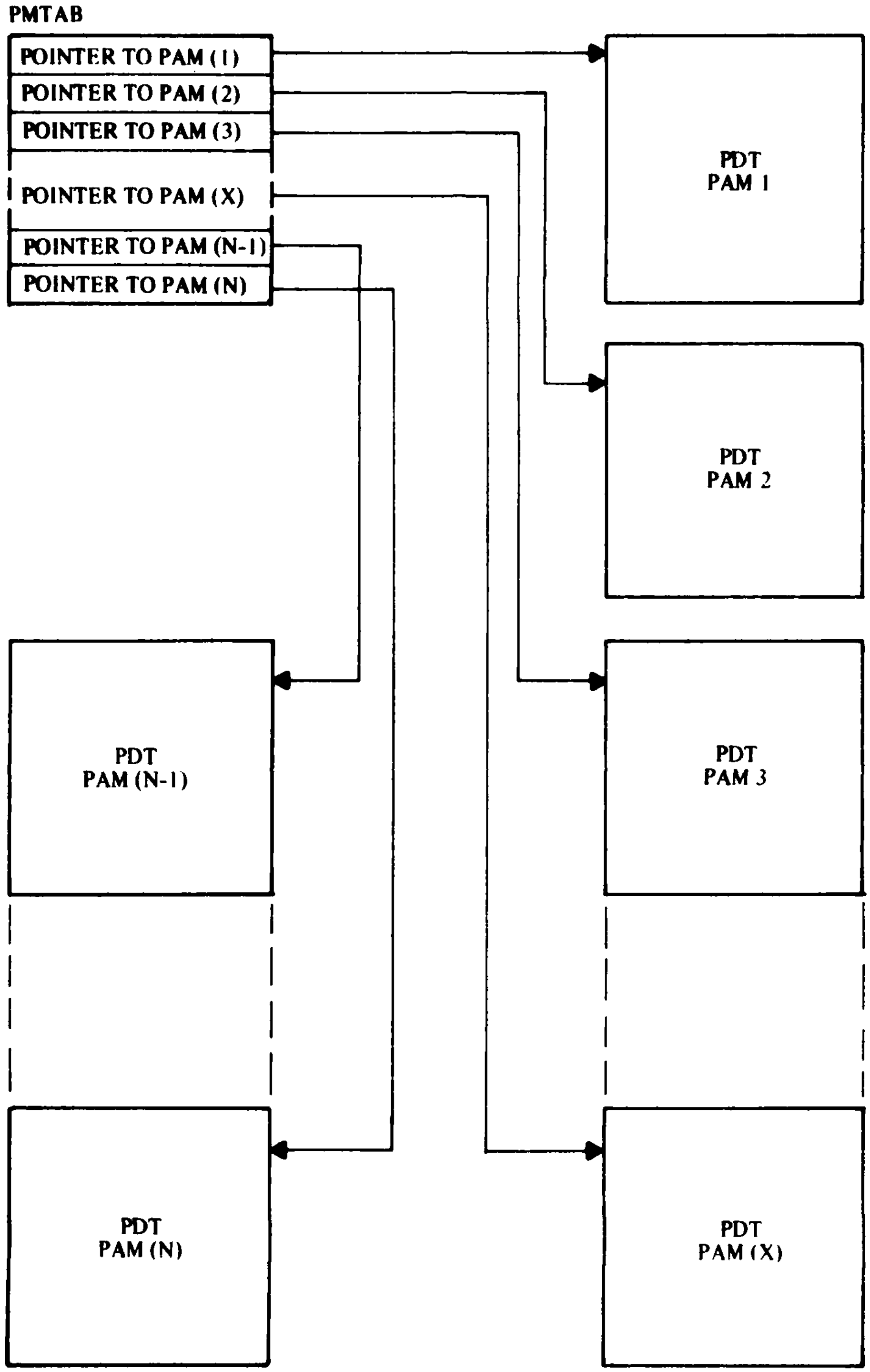


Figure 2-39. Example of Parameter Monitor Setup

5. DISPLAY. If the parameter is to be displayed, the value is placed in the appropriate display buffer and the IDIIOM output routine (IDI:3) is activated.

After all the Parameter Data Tables have been processed, the program outputs to the COGNITRONICS the highest priority message which has been previously selected and then checks to see if the run is to be terminated. If the run is to be terminated, the appropriate termination code is selected, and the appropriate termination message is sent to the COGNITRONICS.

e. Inputs

1. Internal Inputs:

CEXER	Current Exercise Number
CHKTIME	Check Parameter Start Time
DISPT	Display Time Counter
DPINC	Data Processing Time Interval
PMTAB	Parameter Monitor Table
ROUTE	Task Number
RUNTIME	Maximum Run Time
TGPMMIN	Minimum Run Time
TPAMCFG	Required Parameter Terminate Configuration

2. External Inputs: CLOCK F-4 Cycle Time

3. Constants:

ATGATE	Attack Gate Penetration Switch
CMHEAD	Head of COGNITRONICS Message Queue
CPBIT	Check Parameter Bit
DISPTI	Display Time Increment
DSAPBIT	Display Actual Parameter Bit
DSBIT	Display Parameter Bit
D180	180 Degrees
D360	360 Degrees
FS1	1

LOMP	Last Output Message This Category
LOMV	Last Output Message Value
LOMT	Time of Last Output Message This Category
LPWAIT	Line Printer Wait Switch
MASKL8	Lower 8-Bit Mask
MASKL16	Lower 16-Bit Mask
MDTIME	Message Delay Time
PMBIT	Parameter Monitor Bit
PSBIT	Parameter Sample Bit
SCBIT	Parameter Score Bit

f. Outputs

1. Internal Outputs

CNTTAB	Parameter Sample Count
CPAMV	Current Parameter Value
DISPSW	TPM Display Switch
DMSGPTR	Diagnostic Message Pointer
DPTIME	Data Processing Time
GPMABORT	Run Abort Switch
GPMTIME	Run Sample Timer Switch
LASTOUT	Last Output Message
LTMOUT	Last Time Any Message Output
MSGPRI	Message Priority Table
PDISTIME	Parameter Display Timer
RUNTERM	Run Termination Switch
SMTAB	Parameter Sample Table
TERMCODE	Run Termination Code

2. External Outputs: None

g. Program Entrances. BAL, 15 TPM:1

h. Exits. B TPMRET (calling location + 1)

i. Subroutines Called

1. BIFMCK Basic IFM Configuration Limits Check
2. DPOUT Data Processing Output
3. QINSERT COGNITRONICS Output
4. QPURGE Purge COGNITRONICS Queue

j. Memory Requirements

1. Instructions 250
2. Data 7

k. Type of Program Module. ATE/AFT Background

l. Flow Charts. See figure 2-40.

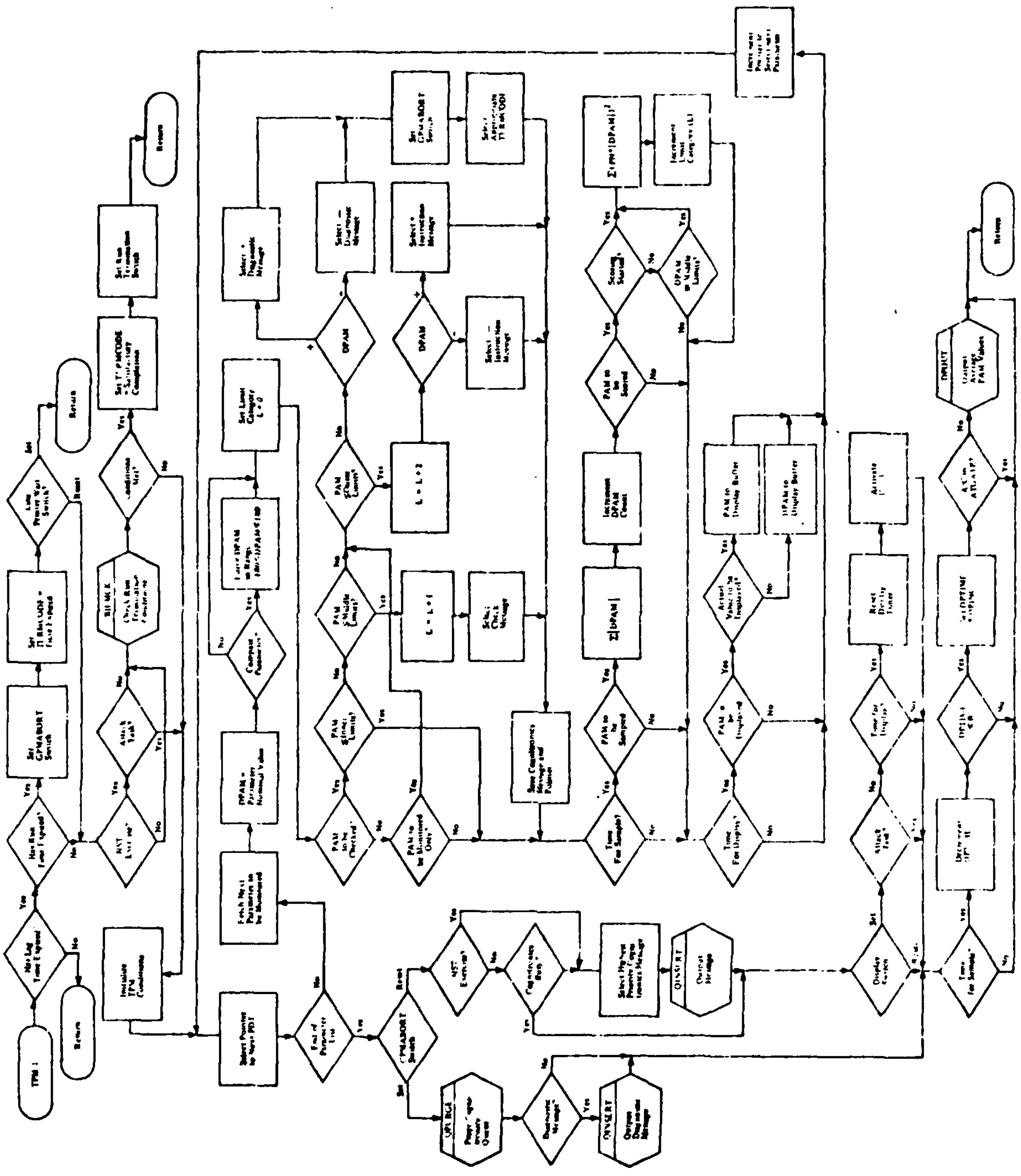


Figure 2-40. Task Performance Monitor

2.22 NAVIGATION/STRIKE TASK TERMINATOR

a. Program Module Name. Navigation Strike Task Terminator (NSTR)

b. Purpose. The purpose of the NSTR module is to terminate the current Navigation/Strike Task and to determine the routing necessary for subsequent tasks.

c. Requirements. The NSTR module is required to:

1. Output to the typewriter and line printer the reason for the run's termination.

2. Increment the run number and total number of runs.

3. Save the status of the current run.

4. Call the Data Processing and Adaptive Logic subroutines, if required.

5. Terminate the flight if:

   (a) The flight time has expired.

   (b) The maximum number of runs per flight is exceeded.

   (c) The maximum difficulty level has been successfully achieved.

6. Select the next task to be performed.

d. Description. During the execution of the various ATE/AFT exercises, selected run status data are retained for use by the NSTR module. Upon completion of the run, these data are evaluated for subsequent program routing and exercise/task selection.

e. Inputs

1. Internal Inputs:

DFTABXMI	Previous Difficulty Table Index
MSTIME	Flight Termination Time
SFNUM	Student File Number
SSTIME	Maximum Flight Time

TERMCODE	Termination Code
TRSKIP	Program Route Bypass Switch

2. External Inputs:

CLOCK	F-4 Cycle Counter
HI	Indicated Altitude

3. Constants:

ILPTR	Pointer to Line Printer Message Table
STMSG	Table of Termination Messages (Typewriter)
TMSGP	Table of Termination Messages (Line Printer)

f. Outputs

1. Internal Outputs:

FLEVL	Student File – Current Difficulty Level
FRUN	Student File – Run Number
FTOTL	Student File – Total Runs
LRUN	Last Run Code
ROUTE	Task Number
SCSW	Session Complete Switch

2. External Outputs:

FCONT3	Flight Control Word

g. Program Entrances. BAL, 15 TSTR:1

h. Exits. B *TSSAV

i. Subroutines Called

1. BIFMCK Basic Configuration Check
2. DPLEG Data Processing for Leg
3. NSAL Navigation/Strike Adaptive Logic

4. NSDP — Navigation/Strike Data Processing

5. POUT:1 — Typewriter Output

j. Memory Requirements

1. Instructions 118
2. Data 4

k. Type of Program Module. Background Program

l. Flow Charts. See figure 2-41.

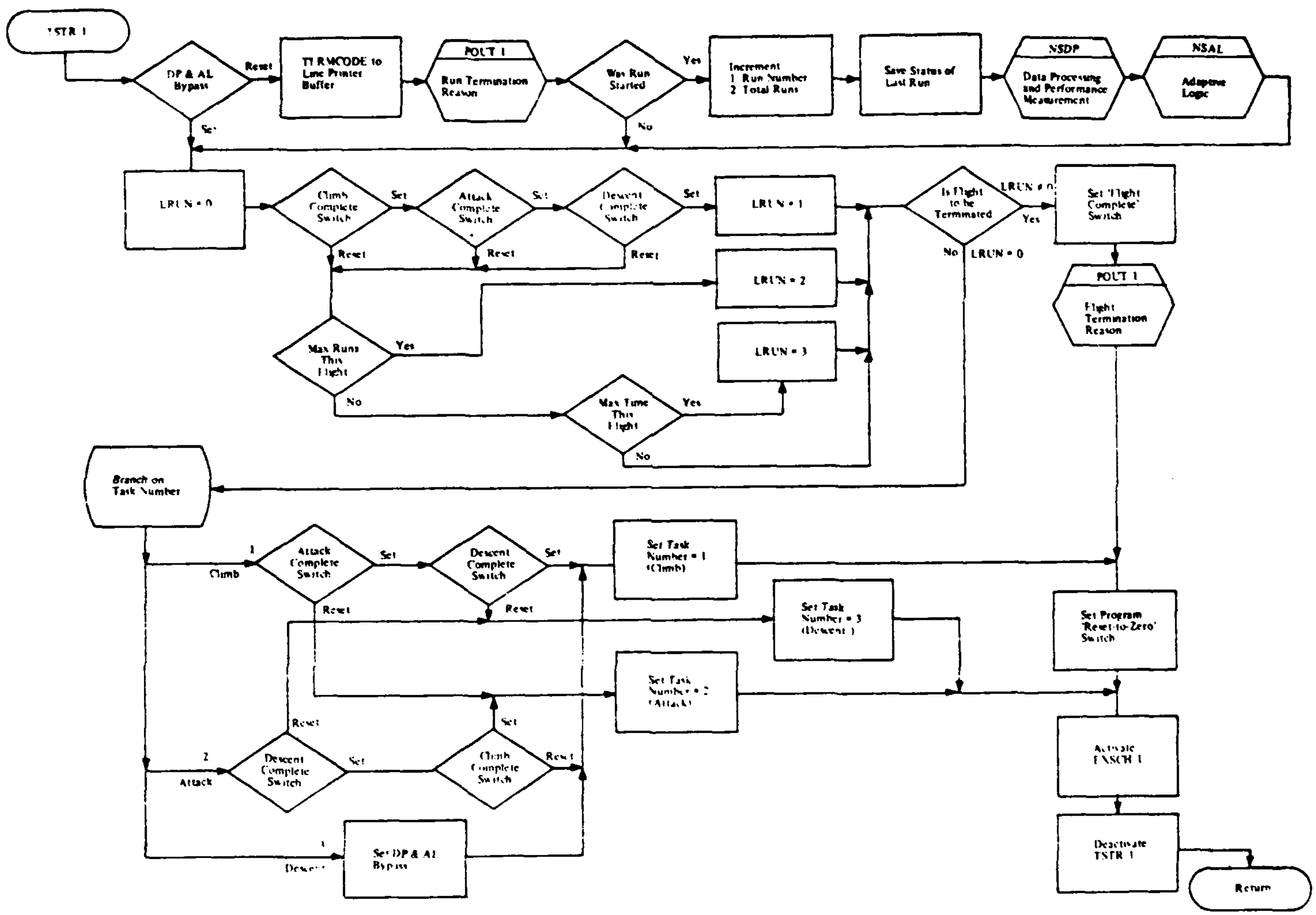


Figure 2-41. Navigation/Strike Task Terminator

2.23 NAVIGATION/STRIKE IDIIOM DISPLAY LIST UPDATE

a. Program Module Name. Navigation Strike IDIIOM Display List Update (NSID)

b. Purpose. The purpose of the NSID module is to initialize and update the Navigation/Strike IDIIOM display lists for the climb, descent, and attack tasks.

c. Requirements. The NSID module is required to:

1. Initialize the climb, descent, and attack display parameters, when directed.

2. Format the display values in IDIIOM absolute vector commands, if it is the first point to be displayed.

3. Compute the difference values of the second and subsequent display parameters and convert them into relative integer raster units.

4. Update the Sigma 7 and IDIIOM display list pointers.

5. Activate the IDIIOM Transmission Routine (IDTR) when data is ready for transmission.

6. Update the student file data in the display list, when requested.

d. Description. The NSID module computations are triggered on a time interval basis from the Task Performance Module (TPM) for the climb and descent tasks, and from the Attack Monitor (NSST) module for the attack task. Currently, the time intervals are 3 seconds for the climb and descent displays and 2 seconds for the attack display. Actual transmission of the display values to the IDIIOM is accomplished via the IDTR module which is activated by this routine. The student file display list buffer is updated by the CLRDISP subroutine contained in this module, but called from the Navigation /Strike Task Selector (NSTS) module.

Three types of IDIIOM transmissions are required:

1. The initial transmission of new student file data to clear any previous data from the display list buffer, and to update the student file display list prior to the start of the next run.

2. The transmission of the first set of data points for each of the values in IDIIOM absolute command formats.

3. The transmission of subsequent data point sets in IDIIOM relative command formats. (Heading is the exception. All heading values are sent in absolute vectors.) The use of relative commands permits a shortened display list and a decreased transmission time.

Figure 2-42 shows a climb task in progress. The display is divided horizontally into sections. The top section is a display of heading displacement with respect to time. The second and third sections are similar displays of altitude and airspeed. The lower section contains the student file data.

The descent task display is similar to the climb task display in that heading, altitude, and airspeed parameters are displayed.

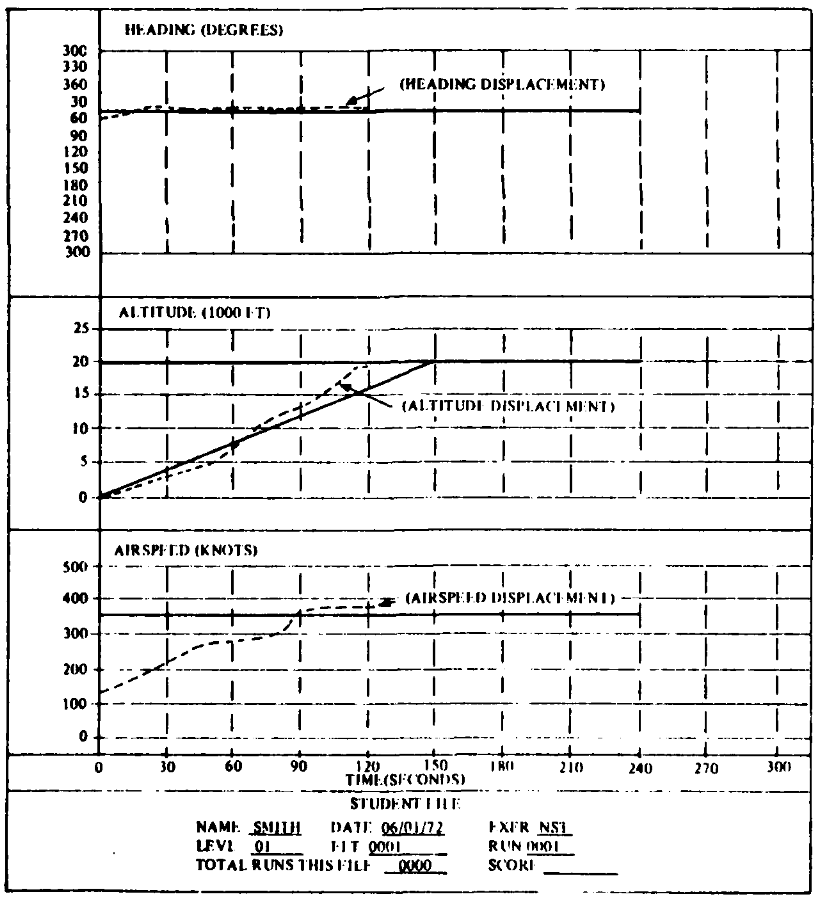


Figure 2-42. Diagram of NST Climb Display

Figure 2-43 shows an attack task in progress. The upper section is a plan view of the attack, while the middle section depicts the altitude displacement from the target aircraft. The lower section again contains file data.

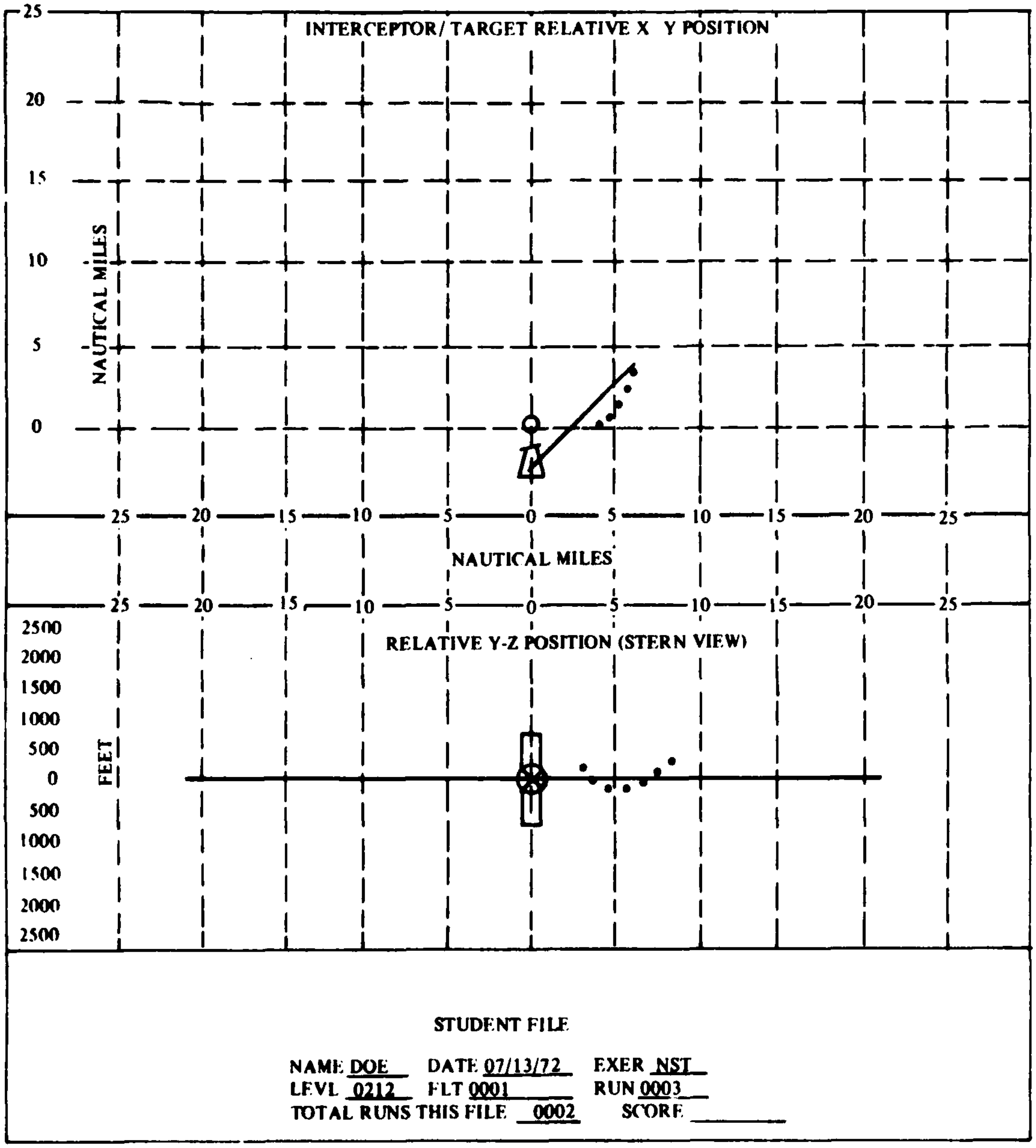


Figure 2-43. Diagram of NST Attack Display

The abscissa of the climb and descent displays represent the length of the run in seconds, and the ordinates indicate the displacement of the parameters from the nominal values. Ordinate scaling depends upon the limits for each of the parameters being displayed. The abscissa scaling is three raster units per second. The abscissa of the NST attack display represents the lateral displacement (nautical miles) of the interceptor from the target. The ordinate in the upper portion of the display represents the longitudinal displacement (nautical miles) of the interceptor from the target while the ordinate in the middle portion of the display represents the vertical displacement (feet) of the interceptor from the target. The scale factors for the attack display are 17 raster units per mile and 0.5 raster unit per foot.

A buffer of I/O command doublewords involving command chaining is utilized for all IDIIOM transmissions. The basic IDIIOM I/O command doubleword format is as follows:

O B	O	BA ( )	
F F	0 0		NB ————
0 1	0	BA (BUF)	
F F	0 0		NB ————
⋮	⋮		⋮
L L L L			

where:

OB is the I/O order for the IDIIOM

BA (   ) is the Sigma 7 byte address of the location containing the starting address in the IDIIOM where data is to be stored.

FF is the I/O command doubleword flags:

02 = no command chaining, suppress incorrect length.
22 = command chaining, suppress incorrect length.

NB ———— is the number of bytes to be transmitted.

BA (BUF) is the Sigma 7 byte address of the first word of the data buffer to be transmitted.

LLLL is the starting address in the IDIIOM where data is to be stored.

An example of the three I/O command doubleword data structures employed during the climb and descent tasks for IDIIOM display list transmission is shown in Figure 2-44. The left hand example shows the I/O command doubleword structure (WRIDI) for transmitting the initial display buffers (NSTHDG, NSTALT, NSTAS, SFDO, SSCORE, NSTDO). At this time, NSTHDG, NSTALT, NSTAS, and SSCORE buffers are filled with NOD (no data) commands which will erase previous data on the IDIIOM screen. The SFDO buffer contains the updated student file data while the NSTDO buffer contains the updated run description data for the next task. The center example depicts the data structure when the first set data points are to be transmitted. The first few bytes of buffers NSTHDG, NSTALT, and NSTAS contain the appropriate line and point mode commands for transmission of the absolute values heading, altitude, and airspeed displacements, while the remainder of the buffer words contain NOD (no data) commands. The right hand example illustrates the data structure for the second data point transmitted. Four bytes are transmitted for NSTHDG, and two bytes each are transmitted for NSTALT and NSTAS. These values replace the next set of NOD commands. Note that the Sigma 7 and IDIIOM buffer addresses have been incremented to correspond to the proper memory locations. Subsequent data point sets are processed in the same manner as the second data point set with both the Sigma 7 and IDIIOM buffer addresses being incremented accordingly.

Figure 2-45 is a similar example of the I/O command doubleword data structures employed during the attack task. The attack display is similar to the climb and descent displays described above.

A portion of the IDIIOM display list data structure for the climb and descent tasks, after the second data point has been transmitted, is shown in Figure 2-46. To convert the parameter displacements to the correct raster coordinates, the following relationships are employed:

For the climb and descent display:

$$TX_{RU} = XINC + XAXIS$$

$$NSTHDG_{RU} = (Value_{NSTHDG})\ (Scale\ Factor_{NSTHDG}) + Offset_{NSTHDG}$$

$$NSTALT_{RU} = (Value_{NSTALT})\ (Scale\ Factor_{NSTALT}) + Offset_{NSTALT}$$

$$NSTAS_{RU} = (Value_{NSTAS})\ (Scale\ Factor_{NSTAS}) + Offset_{NSTAS}$$

INITIALIZATION

WRIDI	1 7	0	BA(C6205A)
	2 2	0 0	0 0 0 2
	1 3	0	BA(C6205B)
	2 2	0 0	0 0 0 2
	0 B	0	BA(C6201)
	2 2	0 0	0 0 0 2
	0 1	0	BA(NSTHDG)
	2 2	0 0	0 1 A C
	0 B	0	BA(C6201)
	2 2	0 0	0 0 0 2
	0 1	0	BA(NSTALT)
	2 2	0 0	0 1 A C
	0 B	0	BA(C6203)
	2 2	0 0	0 0 0 2
	0 1	0	BA(NSTAS)
	2 2	0 0	0 1 A C
	0 B	0	BA(C6204)
	2 2	0 0	0 0 0 2
	0 1	0	BA(SFDO)
	2 2	0 0	0 0 4 C
	0 B	0	BA(C620C)
	2 2	0 0	0 0 0 2
	0 1	0	BA(SSCORE)
	2 2	0 0	0 0 0 C
	0 B	0	BA(C6205)
	2 2	0 0	0 0 0 2
	0 1	0	BA(NSTDO)
	0 2	0 0	0 0 C 0

C6201	0 1 9 F	0 6 0 0
C6202	0 2 7 5	0 6 0 0
C6203	0 3 4 B	0 6 0 0
C6204	0 4 2 1	0 6 0 0
C6205	0 4 4 D	0 6 0 0
C6205A	0 7 A F	0 6 0 0
C6205B	0 0 8 0	0 6 0 0

NSTHDG	F C 0 0	F F 0 0
	F F 0 0	F C 0 0
	F F 0 0	F F 0 0
NSTALT	F F 0 0	F F 0 0
	F C 0 0	F F 0 0
	F F 0 0	F 8 0 0
	F F 0 0	F F 0 0
NSTAS	F F 0 0	F F 0 0
	F C 0 0	F F 0 0
	F F 0 0	F 8 0 0
	F F 0 0	F F 0 0
SFDO	F F 0 0	F F 0 0
	STUDENT FILE DATA	
SSCORE	8 2 F C	9 0 0 D
	F F 0 0	F F 0 0
	F F 0 0	F F 0 0
NSTDO	F F 0 0	F F 0 0
	F F 0 0	F F 0 0
	8 0 0 1	9 1 1 9
	MACH OR KNOTS DISPLAY DATA	
	F 8 0 0	F F 0 0
	CLIMB OR DESCENT DISPLAY DATA	

1ST DATA POINT

WRIDI	0 B	0	BA(C6201)
	2 2	0 0	0 0 0 2
	0 1	0	BA(NSTHDG)
	2 2	0 0	0 0 0 8
	0 B	0	BA(C6202)
	2 2	0 0	0 0 0 2
	0 1	0	BA(NSTALT)
	2 2	0 0	0 0 0 8
	0 B	0	BA(C6203)
	2 2	0 0	0 0 0 2
	0 1	0	BA(NSTAS)
	0 2	0 0	0 0 0 8

C6201	0 1 9 F	0 6 0 0
C6202	0 2 7 5	0 6 0 0
C6203	0 3 4 B	0 6 0 0

NSTHDG	F C 0 0	(ABS TIME)
	(ABS Hdg)	F C 0 0
	F F 0 0	F F 0 0
NSTALT	F F 0 0	F F 0 0
	F C 0 0	(ABS TIME)
	(ABS Alt)	F 8 0 0
	F F 0 0	F F 0 0
NSTAS	F F 0 0	F F 0 0
	F C 0 0	(ABS TIME)
	(ABS A/S)	F 8 0 0
	F F 0 0	F F 0 0
	F F 0 0	F F 0 0

2ND DATA POINT

WRIDI	0 B	0	BA(C6201)
	2 2	0 0	0 0 0 2
	0 1	0	BA(NSTHDG)
	2 2	0 0	0 0 0 4
	0 B	0	BA(C6202)
	2 2	0 0	0 0 0 2
	0 1	0	BA(NSTALT)
	2 2	0 0	0 0 0 2
	0 8	0	BA(C6203)
	2 2	0 0	0 0 0 2
	0 1	0	BA(NSTAS)
	0 2	0 0	0 0 0 2

C6201	0 1 A 4	0 6 0 0
C6202	0 2 7 9	0 6 0 0
C6203	0 3 4 F	0 6 0 0

NSTHDG	F C 0 0	(ABS TIME)
	(ABS Hdg)	F C 0 0
	(ABS Time)	(ABS Hdg)
	F F 0 0	F F 0 0
NSTALT	F F 0 0	F F 0 0
	F C 0 0	(ABS TIME)
	(ABS Alt)	F 8 0 0
	ΔTime ΔAltitude	F F 0 0
NSTAS	F F 0 0	F F 0 0
	F C 0 0	(ABS TIME)
	(ABS A/S)	F 8 0 0
	Δ Time Δ Airspeed	F F 0 0
	F F 0 0	F F 0 0

FC00 = ENTER LINE AND POINT MODE ABSOLUTE
F800 = ENTER LINE AND POINT MODE RELATIVE
FF00 = NO DATA

A/S = AIRSPEED
Alt = ALTITUDE
Hdg = HEADING

Figure 2-44. Example of Three Types of Sigma 7 to IDIIOM Data Transfer for Climb and Descent Display

INITIALIZATION

WRIDI	1 7	0	BA(C620A)
	2 2	0 0	0 0 0 2
	1 3	0	BA(C6208)
	2 2	0 0	0 0 0 2
	0 B	0	BA (C6206)
	2 2	0 0	0 0 0 2
	0 1	0	BA(ATKXY)
	2 2	0 0	0 1 5 0
	0 B	0	BA(C6207)
	2 2	0 0	0 0 0 2
	0 1	0	BA(ATKYZ)
	2 2	0 0	0 1 5 0
	0 B	0	BA(C6208)
	2 2	0 0	0 0 0 2
	0 1	0	BA(SFDO)
	2 2	0 0	0 0 4 C
	0 B	0	BA(C620C)
	2 2	0 0	0 0 0 2
	0 1	0	BA(SSCORE)
	0 2	0 0	0 0 0 C

C6206	0 6 3 3	0 6 0 0
C6207	0 6 D B	0 6 0 0
C6208	0 7 8 3	0 6 0 0
C620A	0 4 A D	0 6 0 0
C620B	0 4 A F	0 6 0 0
C620C	0 4 4 7	0 6 0 0

ATKXY	F C 0 0	8 1 F E
	9 2 4 4	F 8 0 0
	F F 0 0	F F 0 0
	F F 0 0	F F 0 0
ATKYZ	F C 0 0	F F 0 0
	F F 0 0	F 8 0 0
	F F 0 0	F F 0 0
	F F 0 0	F F 0 0
SFDO	STUDENT FILE DATA	
SSCORE	8 2 F C	9 0 0 D
	F F 0 0	F F 0 0
	F F 0 0	F F 0 0

1ST DATA POINT

WRIDI	0 B	0	BA(C6206)
	2 2	0 0	0 0 0 2
	0 1	0	BA(ATKXY)
	2 2	0 0	0 0 1 0
	0 B	0	BA(C6207)
	2 2	0 0	0 0 0 2
	0 1	0	BA(ATKYZ)
	0 2	0 0	0 0 1 0

C6206	0 6 3 3	0 6 0 0
C6207	0 6 D B	0 6 0 0

ATKXY	F C 0 0	8 1 F E
	9 2 4 4	F 8 0 0
	LDFY	LVTX
	F F 0 0	F F 0 0
ATKYZ	F C 0 0	(Absolute Y)
	(Absolute Z)	F 8 0 0
	F F 0 0	F F 0 0
	F F 0 0	F F 0 0

2ND DATA POINT

WRIDI	0 B	0	BA(C6206)
	2 2	0 0	0 0 0 2
	0 . 1	0	BA(ATKXY)
	2 2	0 0	0 0 1 0
	0 B	0	BA(C6207)
	2 2	0 0	0 0 0 2
	0 1	0	BA(ATKYZ)
	0 2	0 0	0 0 1 0

C6206	0 6 3 B	0 6 0 0
C6207	0 6 E 3	0 6 0 0

ATKXY	F C 0 0	8 1 F E
	9 2 4 4	F 8 0 0
	LDFY	LVTX
	(ΔY,ΔX)	F F 0 0
	F F 0 0	F F 0 0
ATKYZ	F C 0 0	(Absolute Y)
	(Absolute Z)	F 8 0 0
	ΔY,ΔZ)	F F 0 0
	F F 0 0	F F 0 0

FC00 ENTER LINE AND POINT MODE ABSOLUTE
F800 ENTER LINE AND POINT MODE RELATIVE
FF00 NO DATA
LDFY LONG DEFERRED (Y) COMMAND
LVTX LONG VECTOR (X) COMMAND

Figure 2-45. Example of Three Types of Sigma 7 to IDIIOM Data Transfer for Attack Display

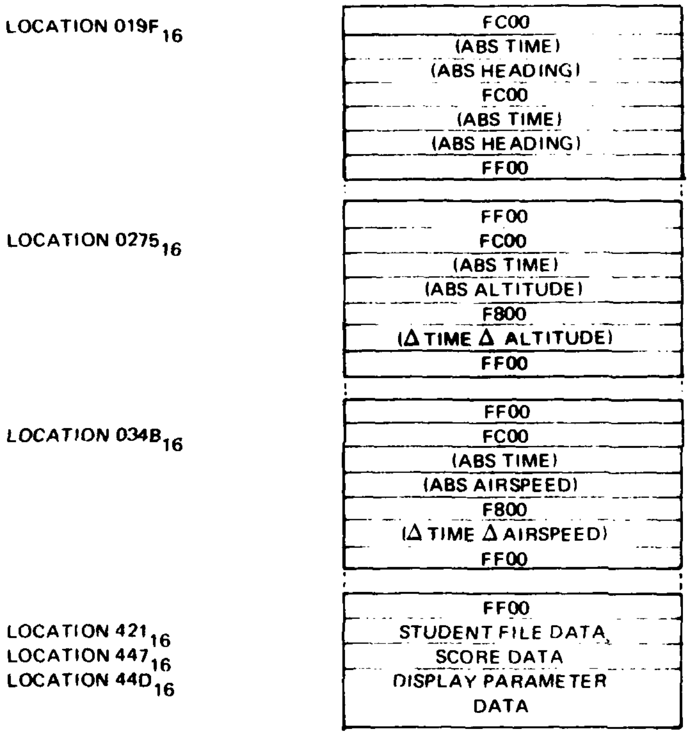


Figure 2-46. IDIIOM Display List Data Structure for Climb and Descent Display after Transmission of Second Data Point (Note: The Attack Display List Structure is similar)

where:

TX_{RU}, $NSTHDG_{RU}$, $NSTALT_{RU}$, and $NSTAS_{RU}$ are the TX, NSTHDG, NSTALT, and NSTAS raster unit coordinates.

$Value_{NSTHDG}$, $Value_{NSTALT}$, and $Value_{NSTAS}$ are the NSTHDG, NSTALT, and NSTAS values.

$Scale\ Factor_{NSTHDG}$, $Scale\ Factor_{NSTALT}$, and $Scale\ Factor_{NSTAS}$ are the NSTHDG, NSTALT, and NSTAS scale factors.

$Offset_{NSTHDG}$ = 800 raster units.

$Offset_{NSTALT}$ = 544 raster units.

$Offset_{NSTAS}$ = 288 raster units.

For the attack display:

$$YIMI_{RU} = (Value_{YIMI})\ (RUMILE) + Offset_{YIMI}$$

$$XIMI_{RU} = (Value_{XIMI})\ (RUMILE) + Offset_{XIMI}$$

$$ZIMI_{RU} = (Value_{ZIMI})\ (RUFEET) + Offset_{ZIMI}$$

where:

$YIMI_{RU}$, $XIMI_{RU}$, and $ZIMI_{RU}$ are the YIMI, XIMI, and ZIMI raster unit coordinates.

$Value_{YIMI}$, $Value_{XIMI}$, and $Value_{ZIMI}$ are the YIMI, XIMI, and ZIMI values.

RUMILE = 17 raster units per mile.

RUFEET = 0.5 raster unit per foot.

$Offset_{YIMI}$ = 510 raster units.

$Offset_{XIMI}$ = 600 raster units.

$\text{Offset}_{\text{ZIMI}}$ = 350 raster units.

e. Inputs

1. Internal Inputs:

FRUN	Student File — Run Number
FSESS	Student File — Session Number
FTOTL	Student File — Total Runs
ROUTE	Task Number
SFNUM	Current Student File Number
XAXIS	Display Time Value
XIMI	Interceptor X-distance from Target (Last Cycle)
YIMI	Interceptor Y-distance from Target (Last Cycle)
ZIMI	Interceptor Z-distance from Target (Last Cycle)

2. External Inputs: None

3. Constants:

ADR3	IDIIOM Doubleword Structure for Climb and Descent Display
ADR4	IDIIOM Doubleword Structure for Attack Display
DCMD	IDIIOM Coded Instruction for Displays
D360	360 Degrees
IDIFIRST	First Point Flag
INIDA1 INIDA2 INIDA3 INIDA4 INIDA5 INIDA5A	IDIIOM Buffer Addresses for Display Values

INIDA5B	IDIIOM Buffer Addresses for Display Values
INIDA6	
INIDA7	
INIDA8	
INIDAA	
INIDAB	
INIDAC1	
INIDAC2	
INIDWR3A	Initial IDIIOM Doubleword Structure for Climb and Descent Display
INIDWR4	Initial IDIIOM Doubleword Structure for Attack Display
LMSG	Line Printer Messages
LMSG2T	
OFFSET	Offset for Display Parameters
RUFEET	Raster Units per Feet Scale Factor
RUMILE	Raster Units per Mile Scale Factor
SMAXADD	Maximum Display Address for Climb and Descent Displays
SMAXADD1	Maximum Display Address for Attack Display
THREHUND	Raster Units for 300 Degrees
XINC	Time Increment along X-Axis

f. Outputs

1. Internal Outputs:

ALTBUF	Altitude Buffer for Climb and Descent Display

ASBUF	Airspeed Buffer for Climb and Descent Display
ATKXY	Attack Display Output Buffers
ATKYZ	
BUFA	
BUFB	Attack Display Buffers
BUFC	
C6201	
C6202	
C6203	
C6204	
C6205	
C6205A	
C6205B	IDIIOM Display Buffer Pointers
C6206	
C6207	
C6208	
C620A	
C620B	
C620C	
HDGBUF	Heading Buffer for Climb and Descent Displays
NSTALT	Altitude Output Buffer for Climb and Descent Displays
NSTAS	Airspeed Output Buffer for Climb and Descent Displays
NSTHDG	Heading Output Buffer for Climb and Descent Displays
SFD0	Student File Display List

SFD1	Student File Display List – Name
SFD2	Student File Display List – Exercise
SFD3	Student File Display List – Level
SFD4	Student File Display List – Flight
SFD5	Student File Display List – Run
SFD6	Student File Display List – Total Runs
SSCORE	Student Score Display File
TX	Time Along X-Axis (Current Value)
TXBUFA	Absolute Buffer for TX
TXBUFR	Relative Buffer for TX
TXLAST	Time Along X-Axis (Previous Value)
WRIDI	Sigma-7 Addresses

2. External Outputs: None

g. Program Entrances

BAL, 15 IDI:3

BAL, 15 CLRDISP

BAL, 15 IDIAT

h. Exits

B * IDIEX

B * EXIT

B *ATRET

i. <u>Subroutines Called</u>

FIX	Converts floating point values to fixed point
HEXASC	Converts hex value to the appropriate ASCII character
HEXTOBCD	Converts hex value to binary coded decimal.

j. <u>Memory Requirements</u>

Instructions	360
Data	67

k. <u>Type of Program Module</u>. Background

l. <u>Flow Charts</u>. See Figures 2-47 through 2-48.

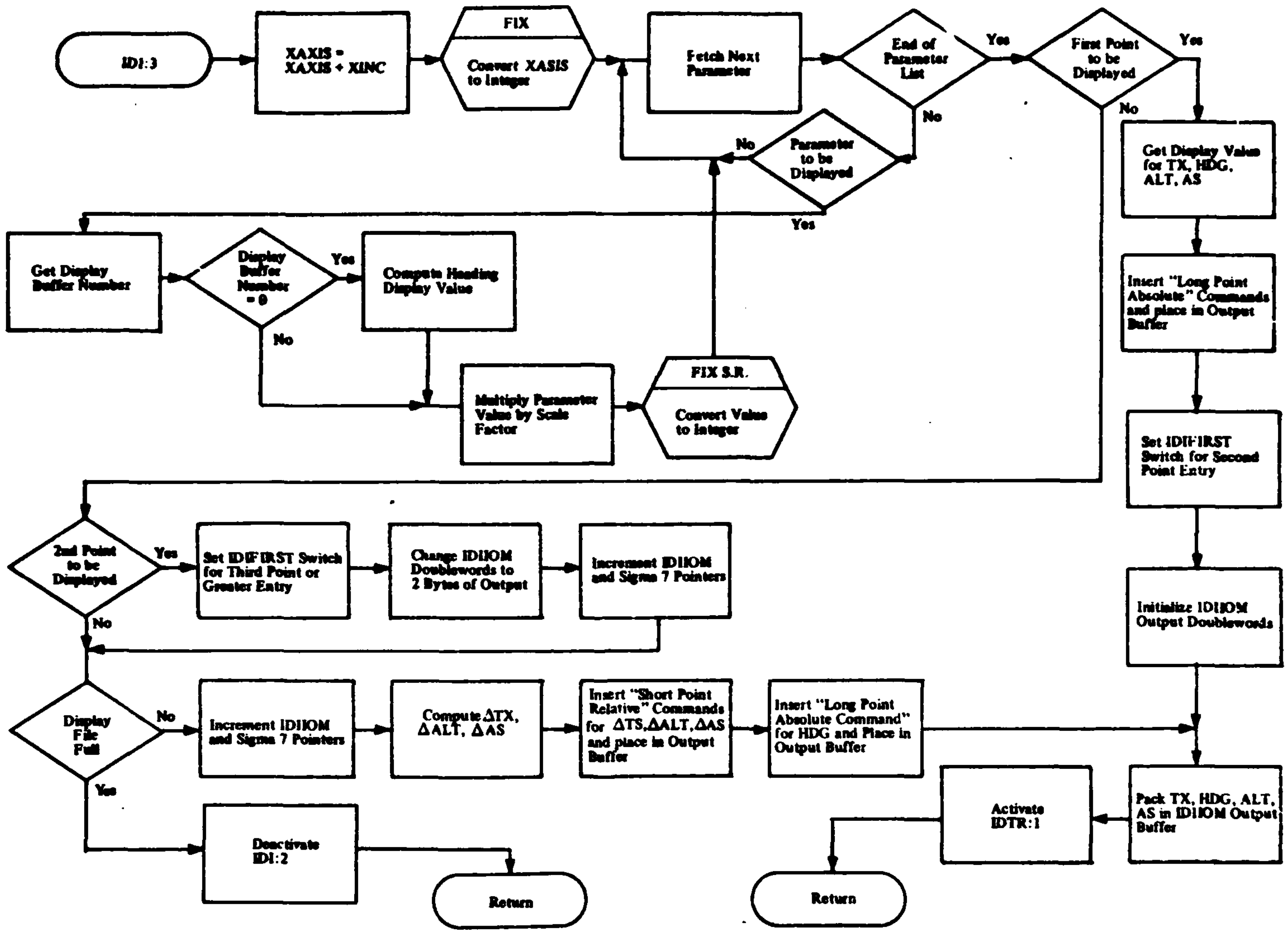


Figure 2-47. NST IDIIOM Display List Update (Sheet 1 of 2)

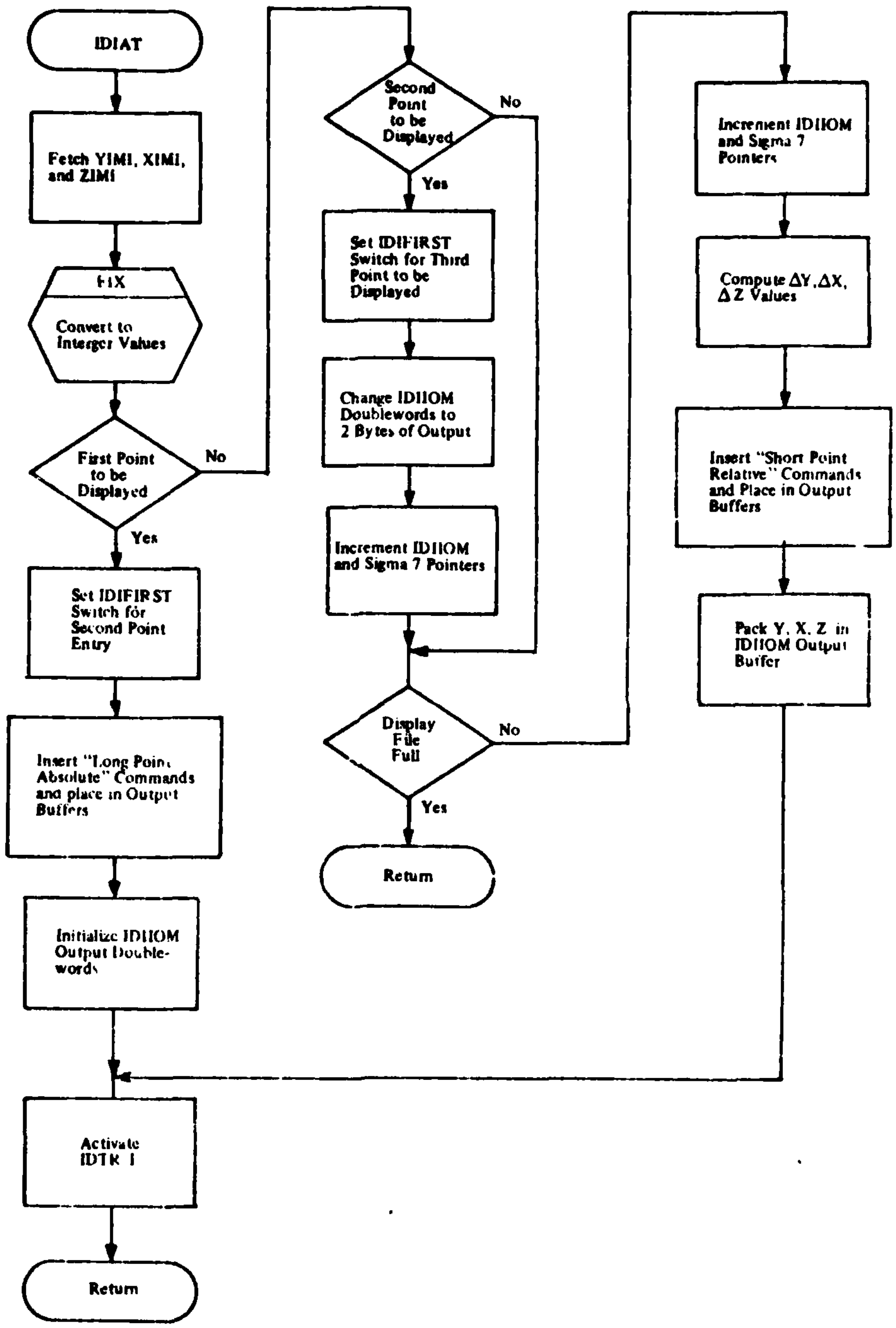


Figure 2-47. NST IDIIOM Display List Update (Sheet 2 of 2)

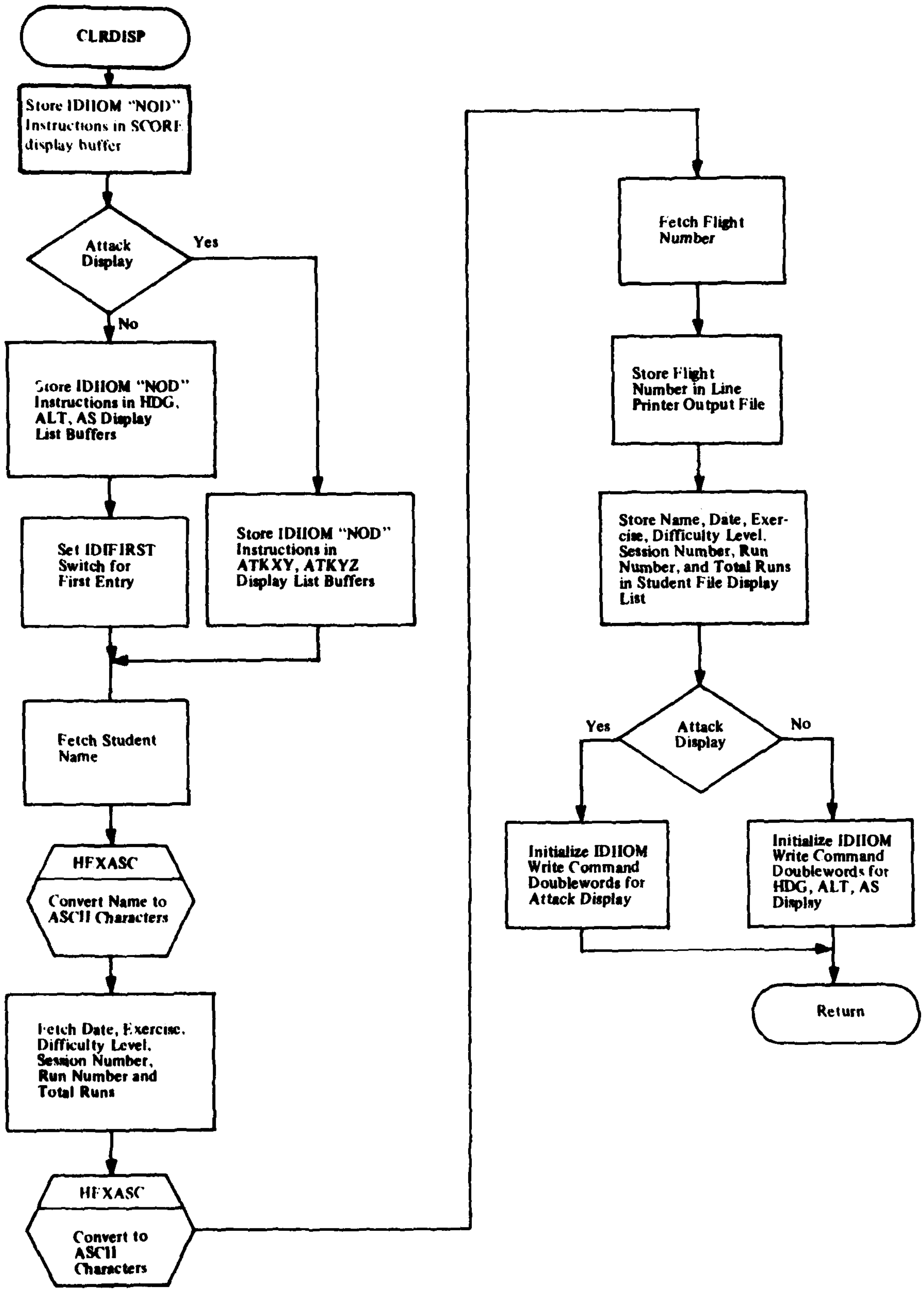


Figure 2-48. CLRDISP Subroutine

2.24 NAVIGATION/STRIKE DATA PROCESSING

a. Program Module Name. Navigation/Strike Data Processing (NSDP)

b. Purpose. The purpose of the NSDP module is to process the performance measurement data for the Navigation/Strike tasks.

c. Requirements. The NSDP module is required to process and output to the line printer the following NST data:

1. General information for the climb, attack, and descent tasks.
2. Parameter variance data.
3. Performance measurement scores.
4. Individual leg scores.
5. Total run score.
6. Run termination data.
7. If an attack task, the gate penetration data, last lock-on data, and attack firing data.

d. Description. The NSDP module consists of the following separate but related routines:

1. NSDP subroutine. This routine is called by the Task Terminator (TSTR:1) module upon completion of the run to compute the performance measurement parameters. Scores are determined as follows:

   (a) Climb or descent task:

$$S = (1.00 + |T_N - T_A|)\left(\frac{\Sigma_1^L L_S}{L}\right)$$

where:

S = Total run score

T_N = Nominal run time

T_A = Actual run time

L_S = Individual leg score (Refer to Paragraph 2-24.d.2 DPLEG subroutine)

L = Number of legs

(b) Attack task:

$$S = A_I + A_F$$

where:

A_I = Score for initial attack leg

A_F = Score for final attack leg

$$A_I = L_S$$

where:

L_S = Individual leg score (Refer to DPLEG subroutine description.)

$$AF = T_{DFR} (T_{DLO} + T_{DFR})$$

where:

T_{DFR} = Target displacement at instant of firing

T_{DLO} = Target displacement at instant of lock-on

If lock-on or firing was not accomplished during the attack run, T_{DFR} and/or T_{DLO} are set to the maximum values.

2. DPLEG subroutine. This routine, called upon completion of each leg, measures the student's performance for that leg. The leg score, L_S, is given as:

$$L_S = \frac{1}{P}\left(\sum_1^P \sqrt{\frac{\Sigma_1^N\left(\frac{|P_C - P_A|}{P_V}\right)^2}{N}}\right)$$

where:

P = Number of parameters monitored

N = Number of samples

P_C = Parameter nominal value

P_A = Parameter actual value

P_V = Normalization factor

3. DPOUT subroutine. This routine outputs one line of parameter average error values to the line printer every 15 seconds during the run, and again at run termination. The actual parameter values are sampled once per second and subtracted from the nominal values contained in the Parameter Description Tables. The results are averaged over a 15-second period in the Task Performance Monitor (TPM) module. This subroutine is called from TPM whenever the 15-second time period expires and again at the termination of a run.

4. DPOUTP subroutine. This routine is called to output either the run entry or run final parameters. It outputs one line of values for the following parameters:

   (a) Altitude (feet)

   (b) Airspeed (knots)

   (c) Airspeed (mach)

   (d) Heading (degrees)

   (e) Climb rate (ft/minute)

   (f) Turn rate (degrees/second)

   (g) Yaw angle (degrees)

   (h) Angle of attack (units)

   (i) Roll angle (degrees)

   (j) Roll rate (degrees/second)

These values can be compared to the original values to check the student's entry and exit conditions.

5. GDOUT and AFDOUT subroutines. These subroutines output the following target data where (1) the final attack gate is penetrated (GDOUT), (2) lock-on is lost (AFDOUT), and (3) missiles are fired (AFDOUT):

   (a) X_I is the distance behind target (nautical miles)

   (b) Y_I is the distance to right of target (nautical miles)

   (c) Z_I is the distance below target (feet)

   (d) Interceptor mach number

   (e) Interceptor heading

   (f) Horizontal look-angle

e. Inputs

1. Internal Inputs:

ADR4	IDIIOM Buffer Address
AFTIME	Final Attack Mode Run Time
CEXER	Current Exercise
CNTTAB	Parameter Sample Count
DRTIME	Desired Run Time
FTOTL	Student File – Total Number of Runs
INIDA4	Initial IDIIOM Address
INIDA8	Initial IDIIOM Address
LA	Horizontal Look-Angle (Relative)
LCOUNT	Line Counter
LEGN	Number of Legs

LTAB	Character Table for 15-second Intervals
PMTAB	Parameter Monitor Table
ROUTE	Task Number
RSTIME	Run Start Time
SFNUM	Student File Number
SINVA	Sine of Vertical Look-Angle
SMTAB	Parameter Sample Table
TDFR	Target Displacement at Time of Firing
TDLO	Target Displacement at Time of Lock-On
TRSKIP	DP and AL Bypass Switch

2. External Inputs:

CLOCK	F-4 Cycle Counter

3. Constants:

ILMSG<u>XX</u>	Line Printer Output Messages
PLP	15 Second Parameter Limit Percent Table
SCBIT	Score Parameter Bit

f. <u>Outputs</u>

1. Internal Outputs:

C620C	IDIIOM Climb and Descent Display Address
C620D	IDIIOM Attack Display Address
LEGSCORE	Leg Score
LTCT	Last Time Counts Transferred
SCORE	Overall Performance Score for Run
SSCORE	IDIIOM Display Output Score

AD-A104 035 LOGICON INC SAN DIEGO CA F/G 5/9
AFT PROGRAM DESCRIPTION NAVIGATION/STRIKE TASKS. PHASF II.(U)
SEP 72 R M JOHNSON N61339-72-C-0108
UNCLASSIFIED SDR-120 NL

3 OF 3
AD
A104 035

END
DATE FILMED
10-81
DTIC

2. External Outputs: None

g. Program Entrances

1. BAL, 15 NSDP

2. BAL, 15 DPLEG

3. BAL, 15 ENTROUT

4. BAL, 15 DPOUTP (Pointer to Line Printer Message in R1)

5. BAL, 15 DPOUT

6. BAL, 15 GDOUT

7. BAL, 15 AFDOUT (Pointer to Output Buffer in R14)

h. Exits

1. B *NSDPRET (calling location + 1)

2. B *LEGSAV (calling location + 1)

3. B *ENTRSAV (calling location + 1)

4. B *SAV (calling location + 1)

5. B *DPSAV (calling location +1)

6. B *GDSAV (calling location + 1)

7. B *AFDSAV (calling location + 1)

i. Subroutines Called

1. D1 Convert Floating Point to EBCDIC

2. FLOAT Convert Integer to Floating Point

3. HEXASC Convert Hexadecimal to ASCII

4. HEXBCD Convert Hexadecimal to EBCDIC

5. LPOUT:1 Line Printer Output

6. SQRTO Compute Square Root

j. Memory Requirements

1. Instructions 394

2. Data 59

k. Type of Program Module. Background

l. Flow Charts. See Figures 2-49 through 2-55.

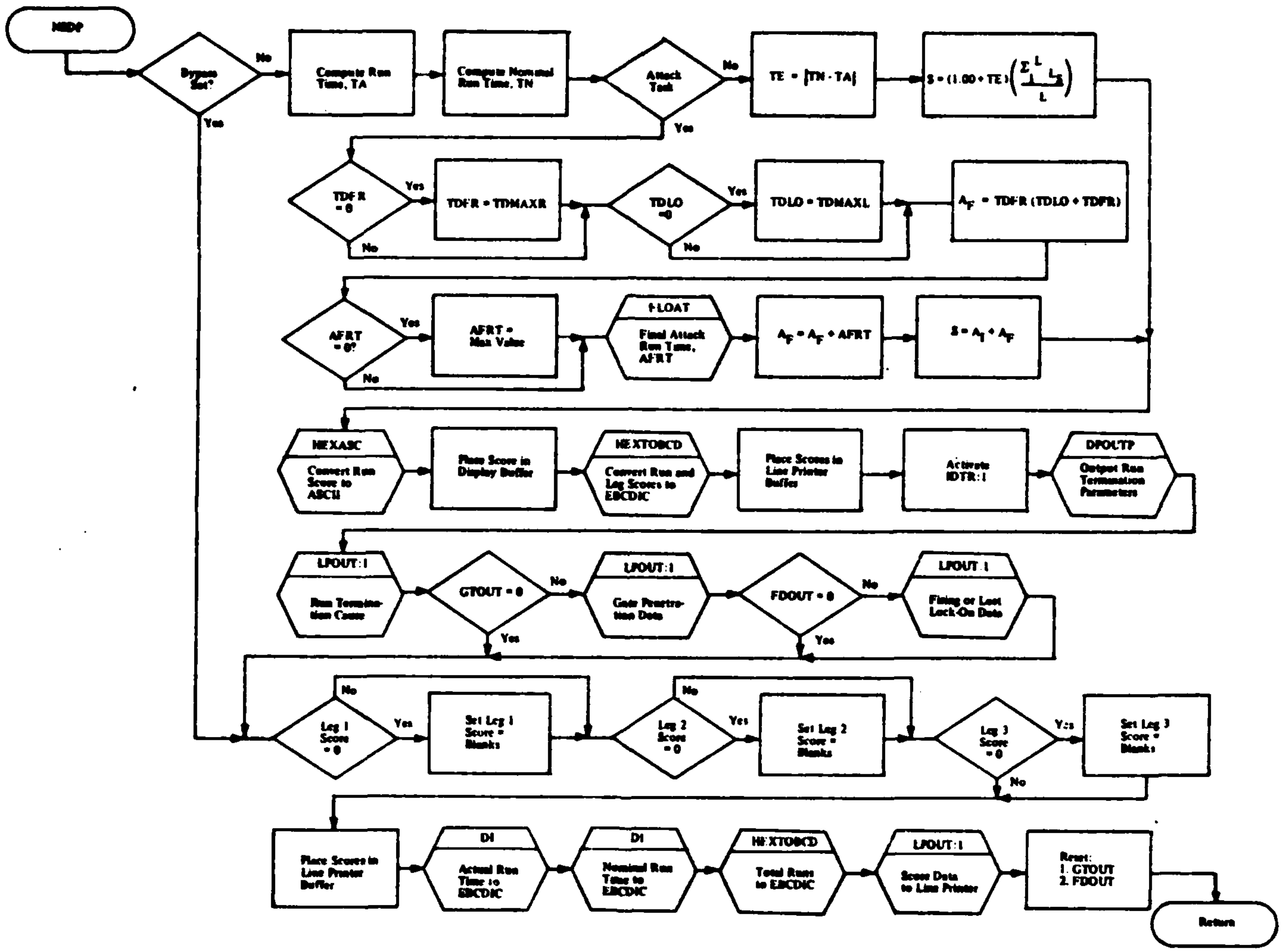


Figure 2-49. Navigation/Strike Data Processing

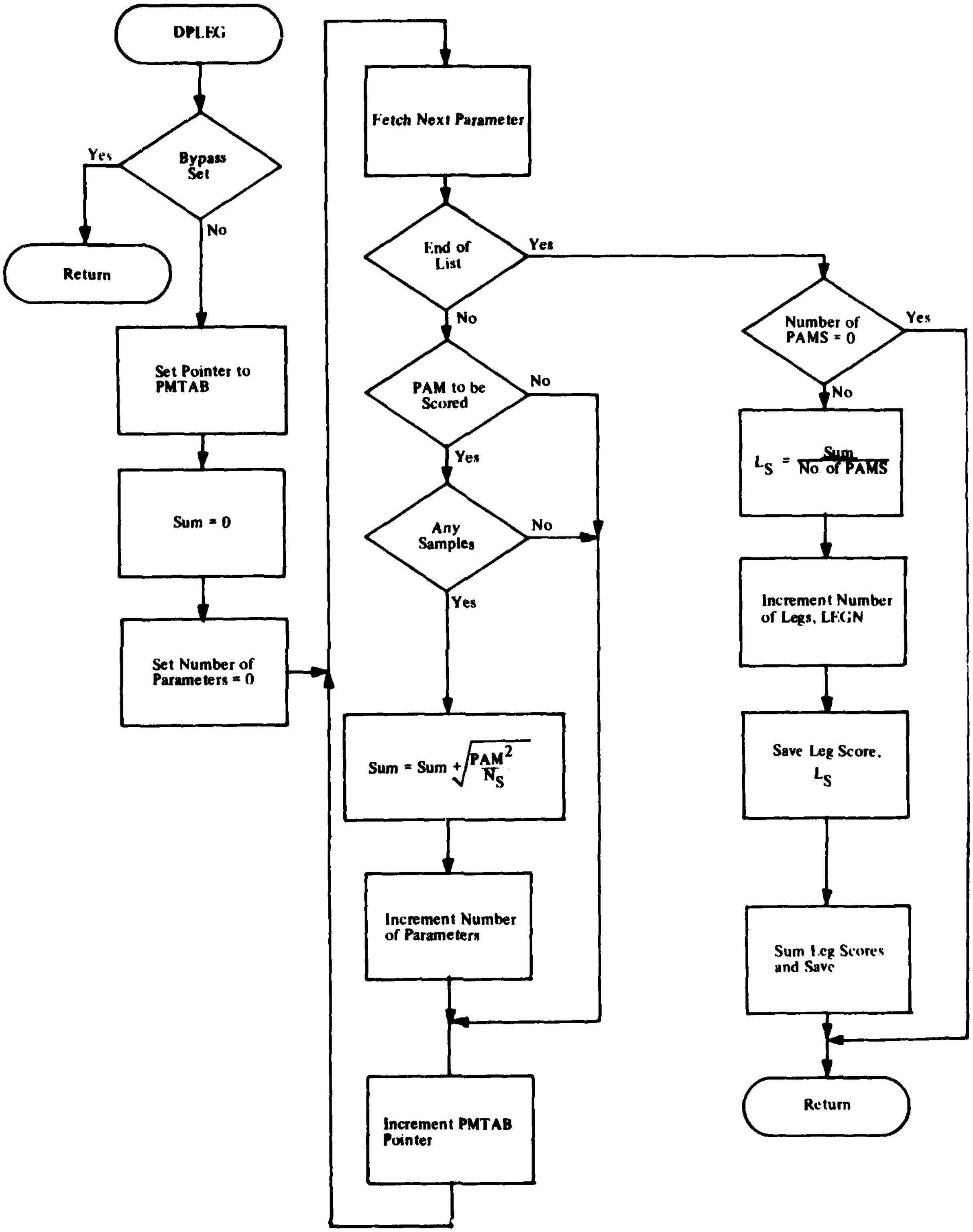


Figure 2-50. Compute Leg Score Subroutine

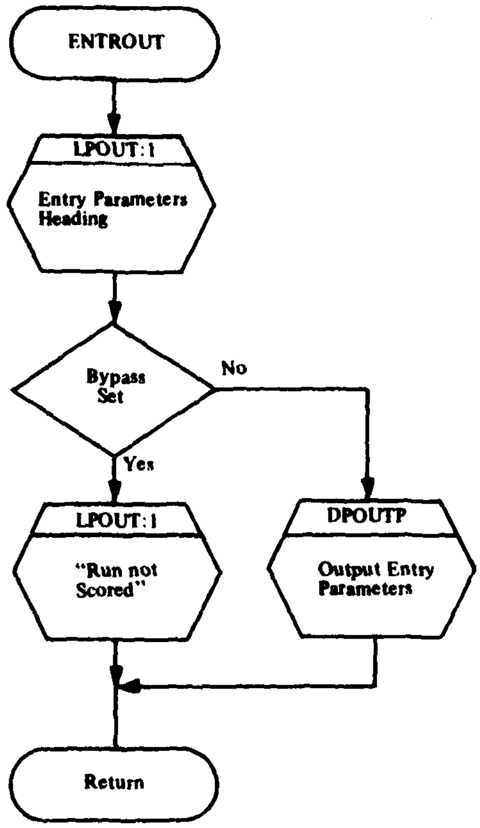


Figure 2-51. Output Run Entry Parameters Subroutine

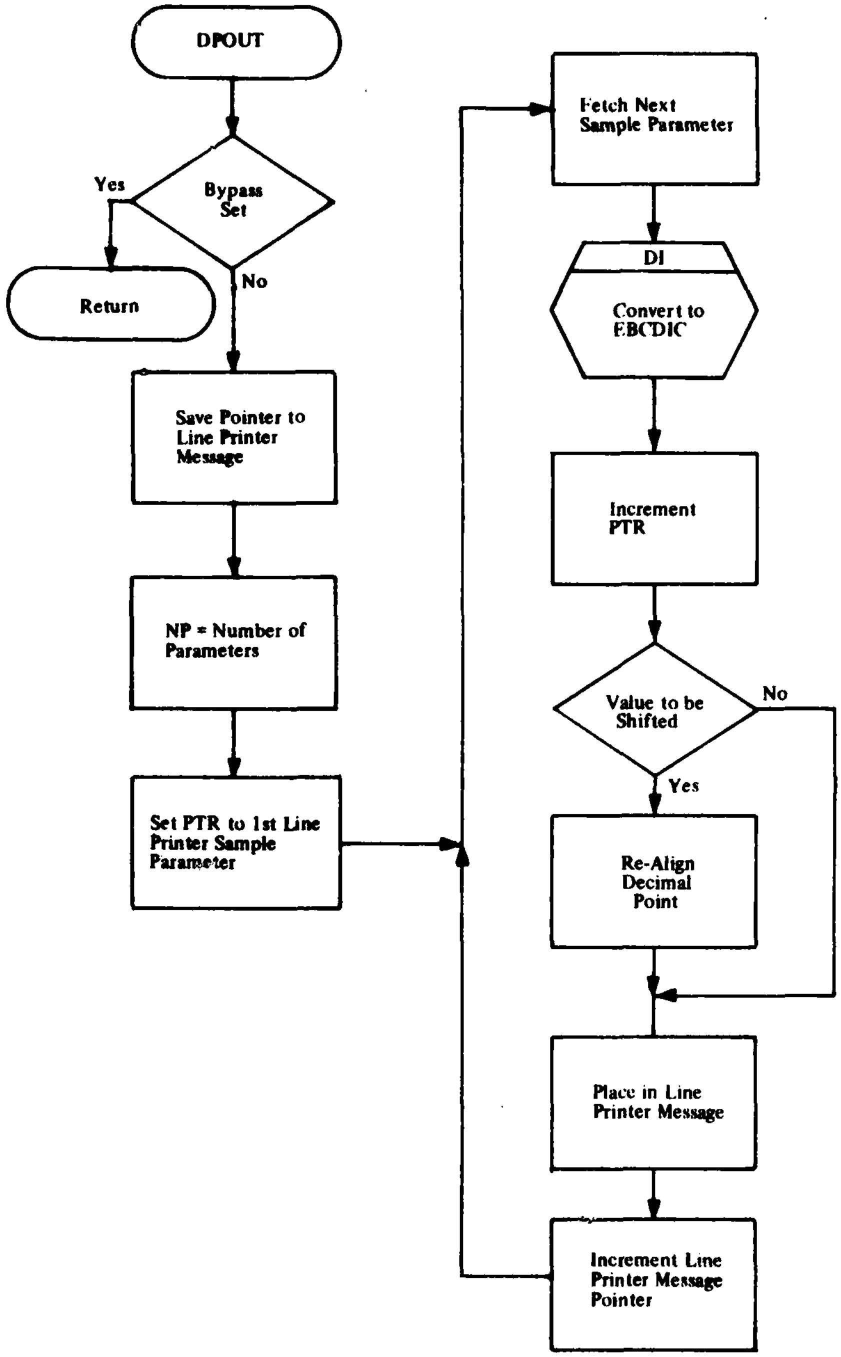


Figure 2-52. Fetch and Save Run Parameters Subroutine

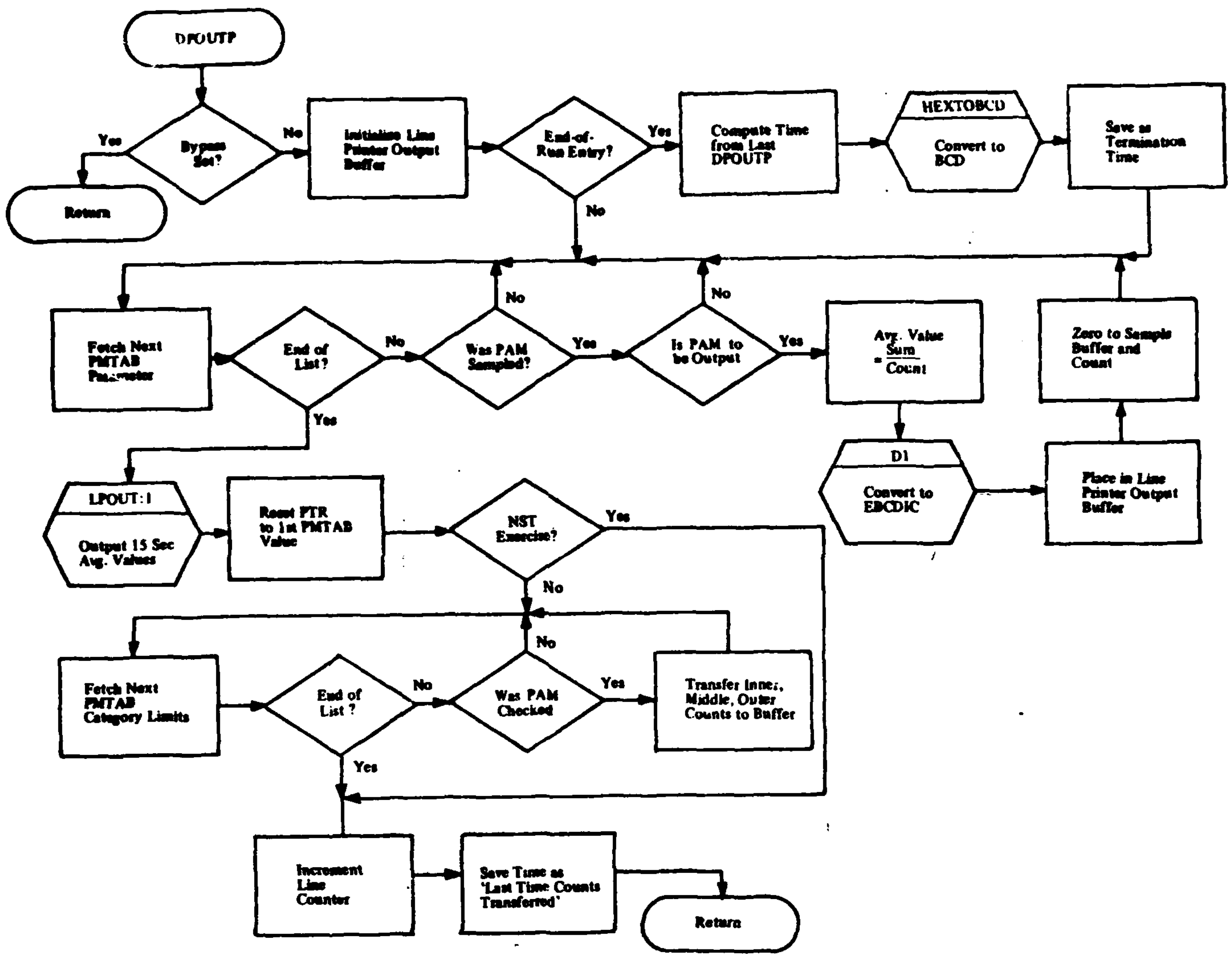


Figure 2-53. Parameter Error Processing Subroutine

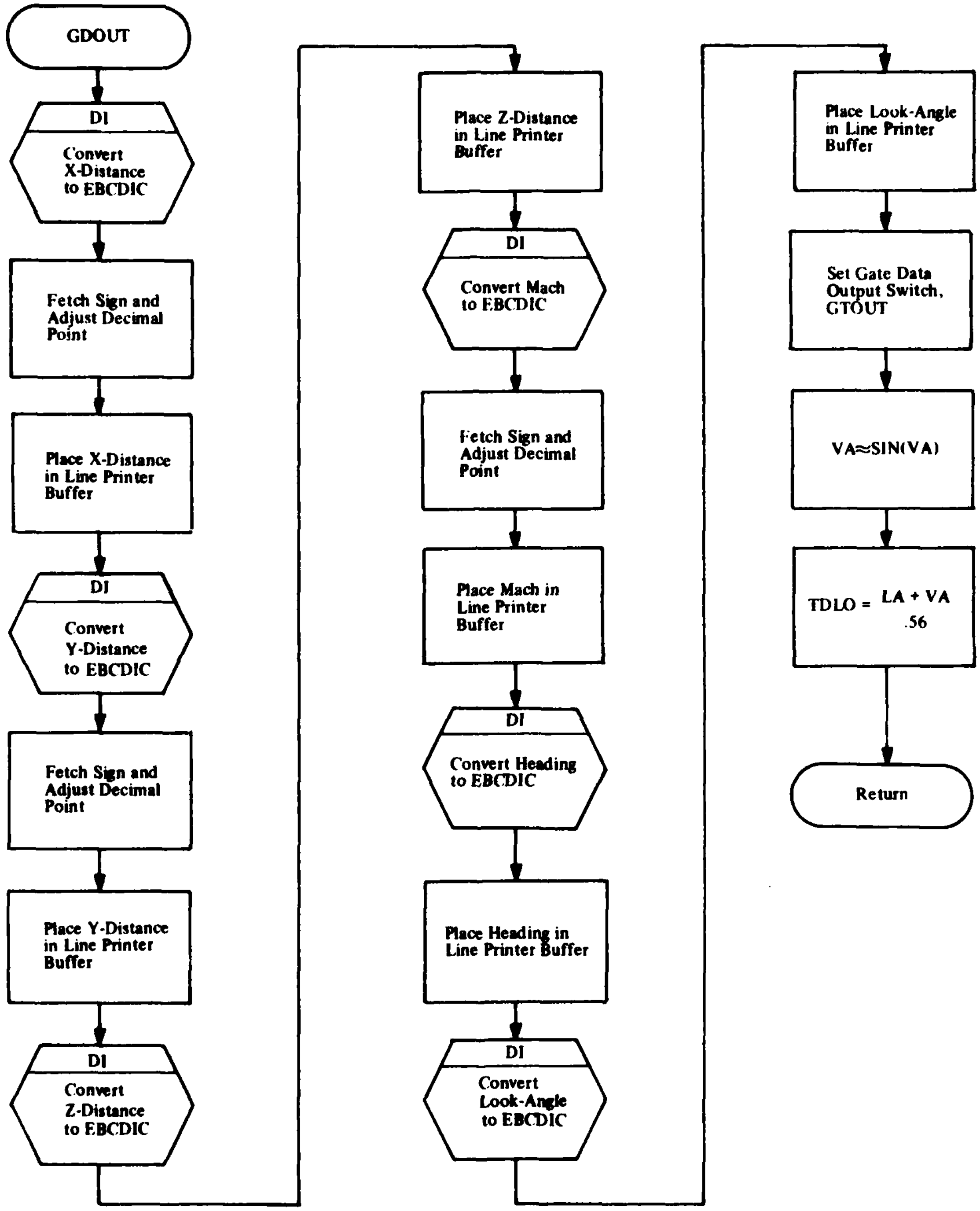


Figure 2-54. Gate Penetration Data Output Subroutine

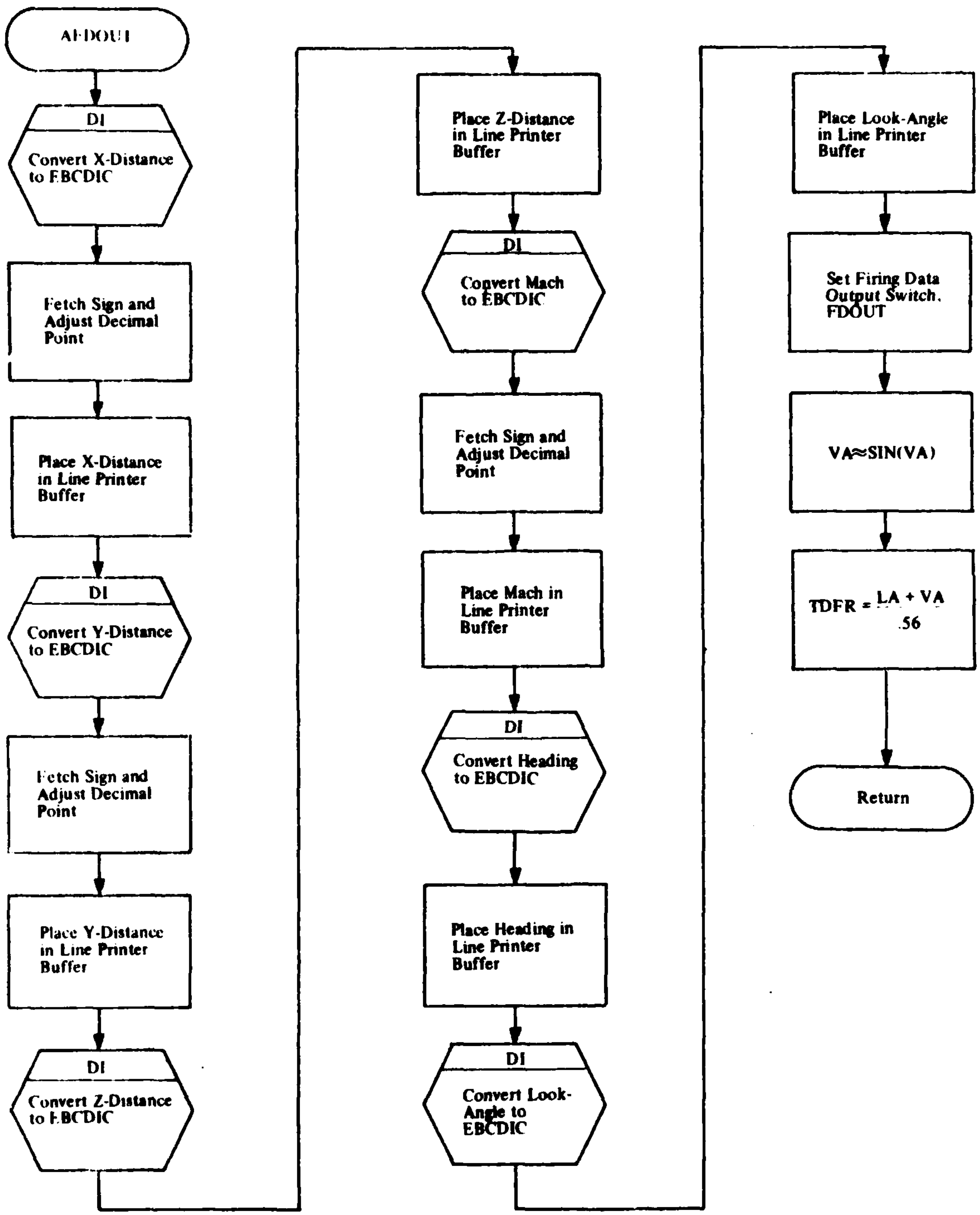


Figure 2-55. Attack Firing Data Output Subroutine

2.25 NAVIGATION/STRIKE ADAPTIVE LOGIC SUBROUTINE

a. Program Module Name. Navigation/Strike Adaptive Logic (NSAL)

b. Purpose. The purpose of the NSAL module is to select and save the difficulty levels to be employed for the next run of the same task (climb, attack, descent) just completed.

c. Requirements. The NSAL subroutine, which is called by the Task Selector (NSTS) module, is required to:

1. Evaluate the subject's performance score for the current run and previous run (same task) if any.

2. Determine the difficulty factor table increment.

3. Select and place in the student file the set of difficulty factors and levels to be employed on the next run, same task (climb, attack, descent).

4. Decode difficulty levels and store the corresponding difficulty factors for subsequent insertion into the AMOD program.

d. Description. The difficulty of subsequent runs is increased or decreased depending on the subject's performance during the past two runs of the same task. Difficulty factors currently employed in NST are center of gravity and turbulence.

Each of these difficulty factors is subdivided into the following levels of difficulty.

1. Center of Gravity (configuration):

   Level 1. 2400 pounds of internal fuel.

   Level 2. Level 1 plus two wing tanks.

   Level 3. Level 2 plus center tank and two Sidewinder missiles.

2. Turbulence:

   Level 1. No turbulence.

   Level 2. 6% of maximum turbulence.

   Level 3. 12% of maximum turbulence.

The student's current task number for each task is retained in the FLEVL table of the Student File and ranges from the simplest combination of climb, attack, and descent tasks to the most difficult. The format used for the FLEVL table is:

0	8	16	24
CN	AN	DN	(spare)

where:

CN = current climb task number (range 1-18)

AN = current attack task number (range 1-18)

DN = current descent task number (range 1-18)

Table NSDLTAB is a table of difficulty levels indexed by the foregoing task numbers, and is arranged in the word relative to the FLEVL table format. The table NSDLTAB format is:

0	4	8	12	16	20	24	28
CCDL	CTDL	ACDL	ATDL	DCDL	DTDL	(spare)	(spare)

where:

CCDL = Climb task configuration difficulty level (range 1-3)

CTDL = Climb task turbulence difficulty level (range 1-3)

ACDL = Attack task configuration difficulty level (range 1-3)

ATDL = Attack task turbulence difficulty level (range 1-3)

DCDL = Descent task configuration difficulty level (range 1-3)

DTDL = Descent task turbulence difficulty level (range 1-3)

The table NSDLTAB difficulty levels currently employed for the Navigation/Strike program are listed in Table 2-10.

Table 2-10. Task Difficulty Level Sequence

Task Number	Climb		Attack		Descent	
	Weight	Turb.	Weight	Turb.	Weight	Turb.
1	1	1	1	1	1	1
2	1	2	1	2	1	2
3	2	2	1	3	2	2
4	2	3	2	1	2	3
5	3	2	2	2	3	2
6	3	3	2	3	3	3
7	1	1	1	1	1	1
8	1	2	1	2	1	2
9	2	2	2	2	2	2
10	2	3	2	3	2	3
11	3	2	3	2	3	2
12	3	3	3	3	3	3
13	2	1	2	1	2	1
14	2	2	2	2	2	2
15	2	3	2	3	2	3
16	3	1	3	1	3	1
17	3	2	3	2	3	2
18	3	3	3	3	3	3

Each trainee is started at the lowest task number for each task and, depending upon his performance, the task number is incremented or decremented by a value from 0 to +3, or from 0 to -3. Task number for each task cannot be incremented beyond the maximum value (18). Task numbers are incremented or decremented by a value determined from the trainee's performance task score (S_T) on the current run, and whether the previous run was incremented, decremented, or remained the same. The technique used to select the sequence number increment is presented in Table 2-11.

Table 2-11. Adaptive Logic Summary Table for Selection of Next Task Number

Previous Run was	$S_T > 300$	$300 \geq S_T \geq 225$	$225 \geq S_T \geq 150$	$150 \geq S_T \geq 75$	$S_T \leq 75$
Decremented (-)	-3	-2	0	0	+1
Not changed (0)	-2	-1	+1	+1	+2
Incremented (+)	-1	0	+1	+2	+3

The NSDF subroutine, although physically a part of the NSAL program module, is called prior to the start of each task by the NPTO module for the climb task, the NSST module for the attack task, and the NSDE module for the descent task. The NSDF routine decodes the next difficulty level and selects the appropriate difficulty factors for the run.

e. Inputs

1. Internal Inputs:

ALCGC	Center Tank Configuration
ALCGF	Internal Fuel Configuration
ALCGS	Sidewinder Missile Configuration
ALCGW	Wing Tank Configuration
ALRA	Rough Air Configuration
DFNST	NST Difficulty Factor Level Limits
NSDLTAB	Climb, Attack, Descent Difficulty Level Table
SCATI	Scoring Categories for Difficulty Factor Increments
SCORE	Total Score
SFNUM	Student File Number
TRSKIP	Program Route Bypass Switch

2. External Inputs: None

3. Constants: None

f. Outputs

1. Internal Outputs:

DFINC	Difficulty Factor Increment
DFTABXMI	Student File – Previous Difficulty Table Index
FLEVL	Task Number Table
NATECT	Center Tank Configuration for Next Run
NATEF	External Fuel Configuration for Next Run
NATEM	Sparrow Missile Configuration for Next Run
NATERAF	Rough Air Factor for Next Run
NATESW	Sidewinder Missile Configuration for Next Run
NATEWT	Wing Tank Configuration for Next Run
ROUTE	Task Number

2. External Outputs: None

g. Program Entrances

BAL, 15 NSAL

BAL, 15 NSDF

h. Exits

B *NSALRET (calling location +1)

B *15 (calling location +1)

i. Subroutines Called. None

j. Memory Requirements

1. Instructions 73

2. Data 11

k. Type of Program Module: Subroutine

l. Flow Charts. See figures 2-56 and 2-57.

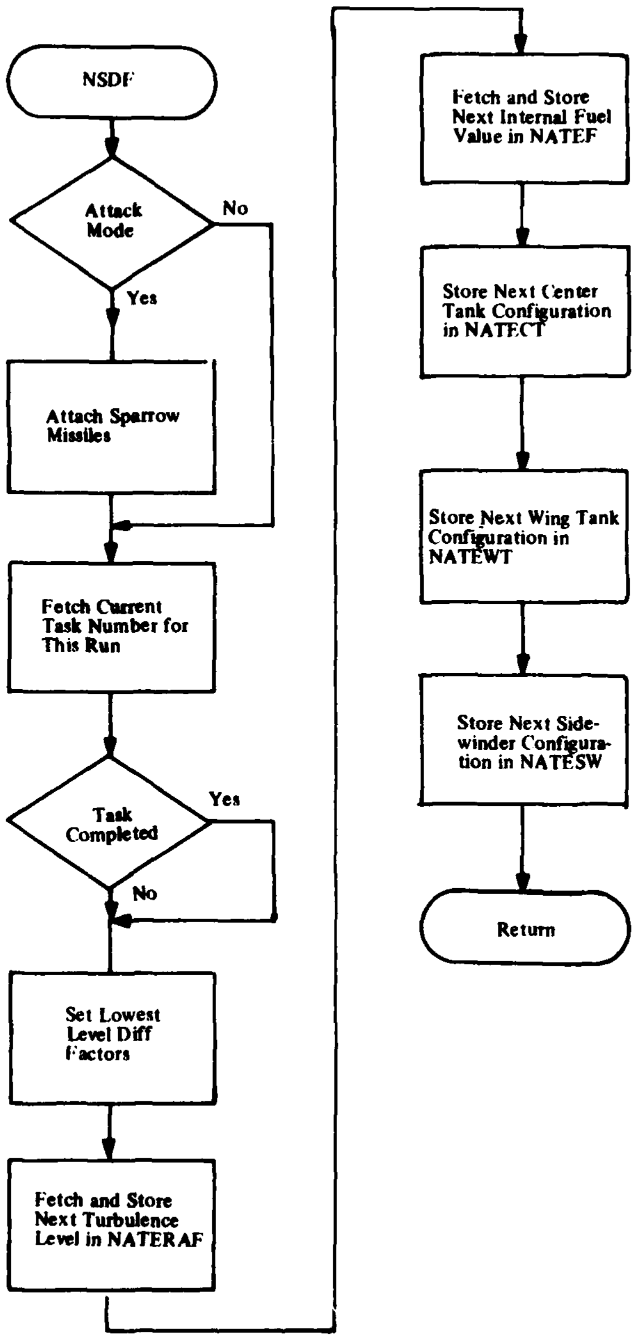


Figure 2-56. NSDF Subroutine

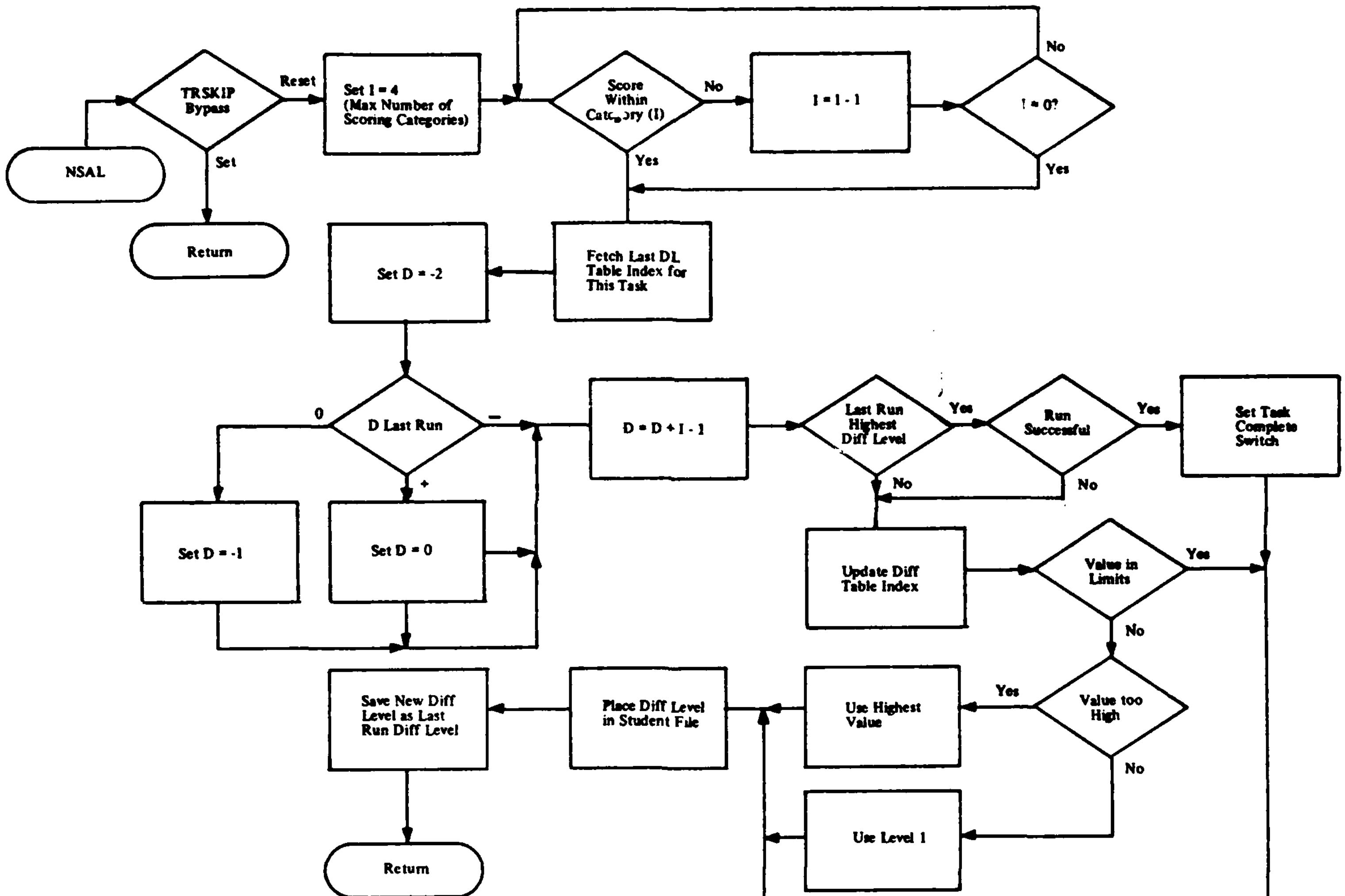


Figure 2-57. NST Adaptive Logic

2.26 STEERING DOT MONITOR

a. Program Module Name. Steering Dot Monitor (SDM)

b. Purpose: The purpose of the SDM module is to monitor and control the simulated steering dot display employed during the final leg of the attack task.

c. Requirements. The SDM module is required to:

1. Provide the student with a simulated visual display of the target relative to the interceptor.

2. Initialize and periodically update the steering dot display data.

3. Terminate the attack task if predetermined criteria are met.

d. Description. Whenever the interceptor enters the attack gate, the SDM module is activated and the simulated steering dot display is initiated. The SDM module first checks for run time expiration (the student is allowed 30 seconds from the time lock-on is achieved to fire the missiles) or if any parameter required to retain lock-on is out of limits. If the final leg is to be terminated, a 'BREAK-"X" ' is displayed on the screen, and voice instructions are issued to the student to break off the attack. If, however, the run is to continue, the target's relative position data is updated and displayed. The system is then interrogated to see if the student has fired the missiles (trigger on the stick depressed) and, if so, the program is placed into an arbitrary 5-second "wait" phase to allow the missiles time to reach the target. At this point, if the target is "hit," it will vanish from the screen. A "hit" is determined if the "look angle" is within 5 degrees at the instant of firing. Six seconds after firing, a 'BREAK-"X" ' is displayed regardless of whether or not the target is hit, and the attack task is terminated.

e. Inputs

1. Internal Inputs:

ILMSG16	Line Printer Message for Attack Firing Data
MACH	Mach Number
MISTIME	Missile Run Time

RUNTERM	Run Termination Switch
SINLOOK	Sine of Look Angle
SINVA	Sine of Vertical Look Angle
TERMCODE	Run Termination Code
T99DI3	F-4 Discrete Input Number Three
XI	Interceptor Distance Behind Target

2. External Inputs:

CLOCK	F-4 Cycle Time

3. Constants: None

f. Outputs

1. Internal Outputs:

ATEM	Sparrow Missiles Atttached Switch
BKTIME	SDM Break Display Time
FIRSW	Firing Switch
GPMABORT	Run Abort Switch
KILLSW	Target Kill Switch
SDMTERM	Attack Completed Switch
SDPHASE	Steering Dot Display Phase

2. External Outputs: None

g. Program Entrances

BAL, 15 SDM:1

h. Exits

B *SDMRET (calling location +1)

i. Subroutines Called

AFDOUT	Attack Firing Data Out Subroutine
FIX	Floating Point to Fixed Point Subroutine

j. Memory Requirements

1. Instructions 232
2. Data 337

k. Type of Program Module. Background

l. Flow Charts. See figures 2-58 through 2-61.

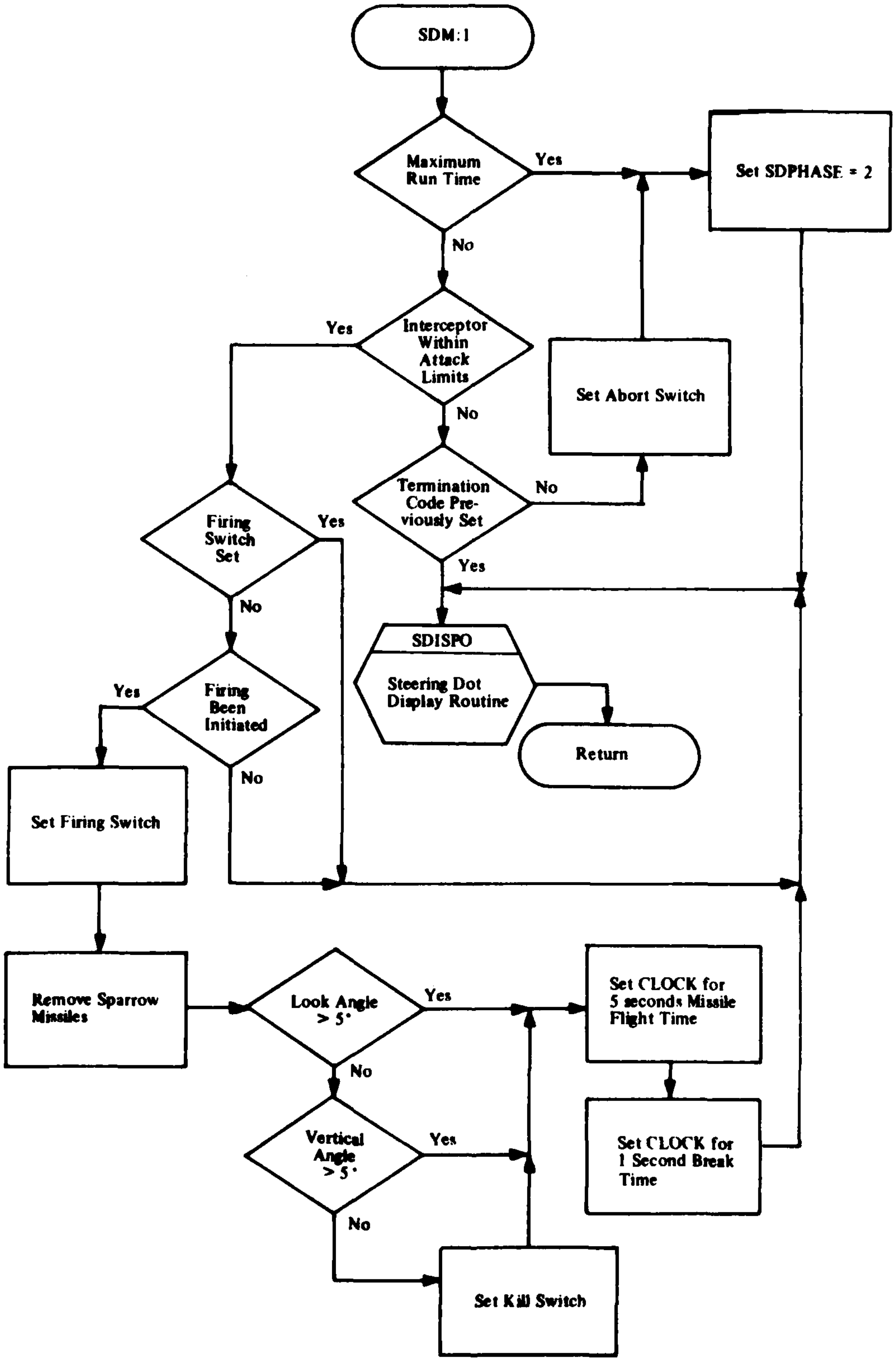


Figure 2-58. Steering Dot Monitor

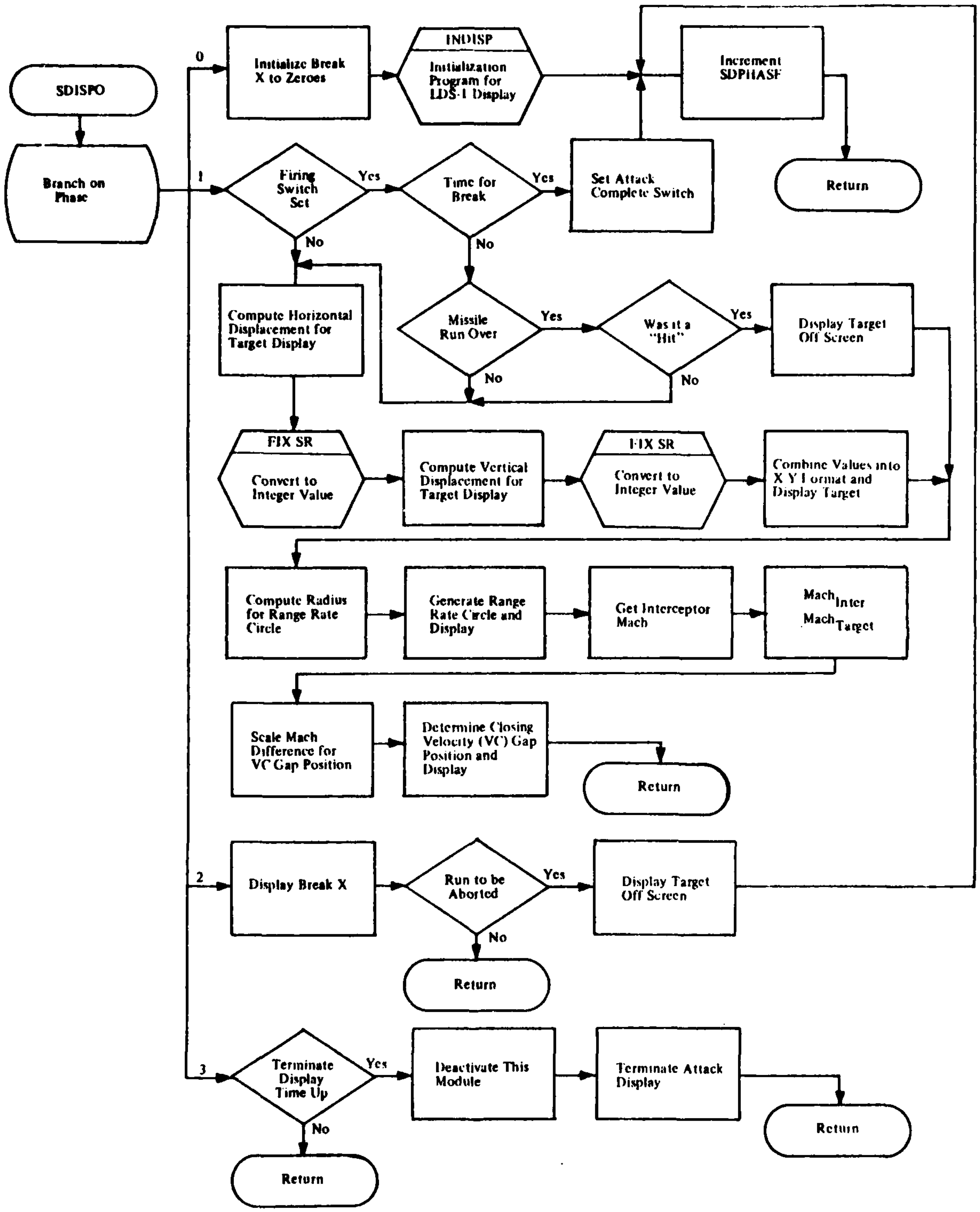


Figure 2-59. Steering Dot Monitor Display Routine

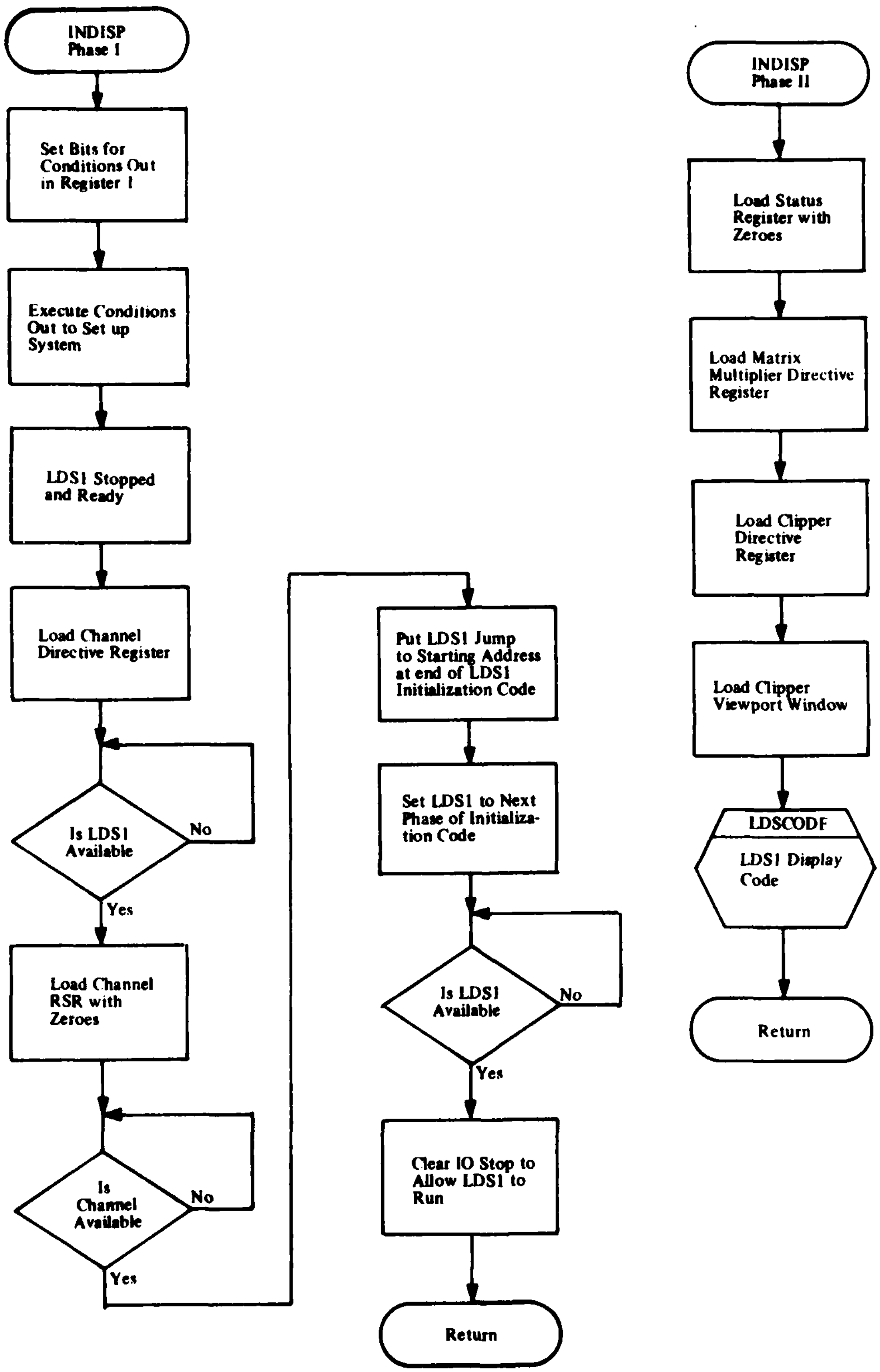


Figure 2-60. Display Initialization Routine

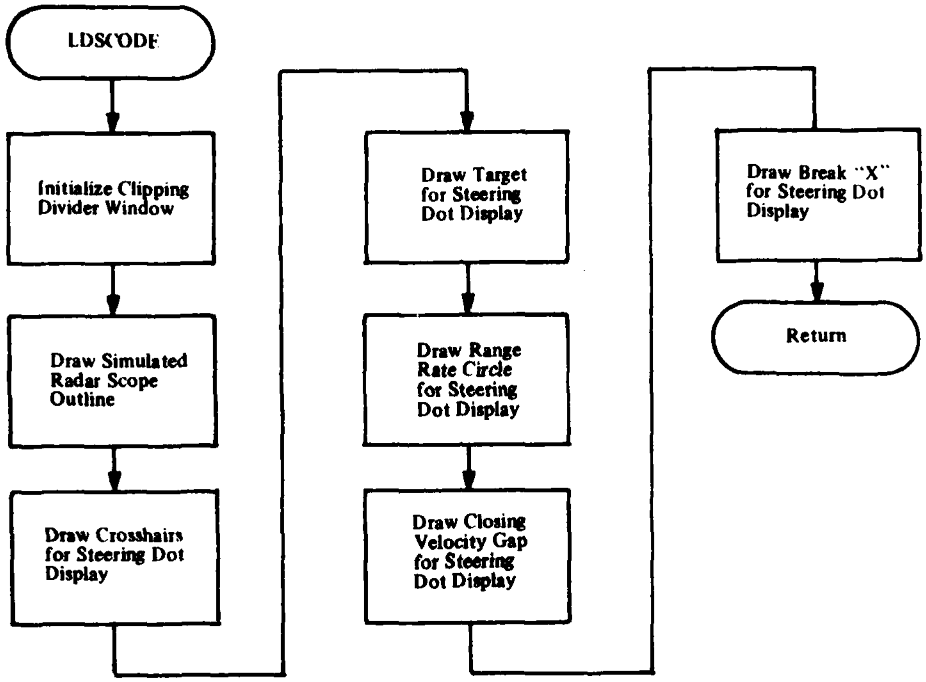


Figure 2-61. Steering Dot Monitor Code for E&S Display

2.27 COGNITRONICS ADDRESSES

a. Program Module Name. Convert Floating Point to COGNITRONICS Addresses (CADR)

b. Purpose. The purpose of the CADR subroutine is to convert a floating point number to the appropriate integer used to address the COGNITRONICS voice drum.

c. Requirements. The CADR subroutine is required to do the following:

1. Check to ensure the floating point number to be converted is in the range 0-999.

2. Obtain the hundreds, tens, and units integer COGNITRONICS addresses from the floating point number.

d. Description. The CADR subroutine is employed in the ATE/AFT program to convert a floating point number in the range 0-999 to three COGNITRONICS addresses used to address the voice drum. The hundreds, tens, and units values of the floating point number are obtained by counting the number of successive subtractions needed to reduce the number to a negative value. Each of the three resultant integers is then used as an index to select the corresponding COGNITRONICS address from a predefined table (COGTAB). If the floating point value is outside the range 0-999, three silence words will be returned.

e. Inputs

1. Internal Inputs: The only internal input is the floating point value which must be in register 14 upon entry.

2. External Inputs: None

3. Constants:

COGLIM	Floating point limits for input value.
COGTAB	A predefined table for converting decimal integers to COGNITRONICS addresses.

f. Outputs

1. Internal Outputs: The following COGNITRONICS addresses will be in the indicated registers upon exit:

COGADR (hundreds value)	Register 1
COGADR (tens value)	Register 2
COGADR (units value)	Register 3

2. External Outputs: None

g. Program Entrances

BAL, 15 COGADDR

h. Exits

B *15 (calling location +1)

i. Subroutines Called. SELECT. This is an internal subroutine which selects COGNITRONICS addresses from table COGTAB.

j. Memory Requirements

1. Instructions 33
2. Data 17

k. Type of Program Module. Subroutine callable from ATE Background programs.

l. Flow Charts. See figure 2-62.

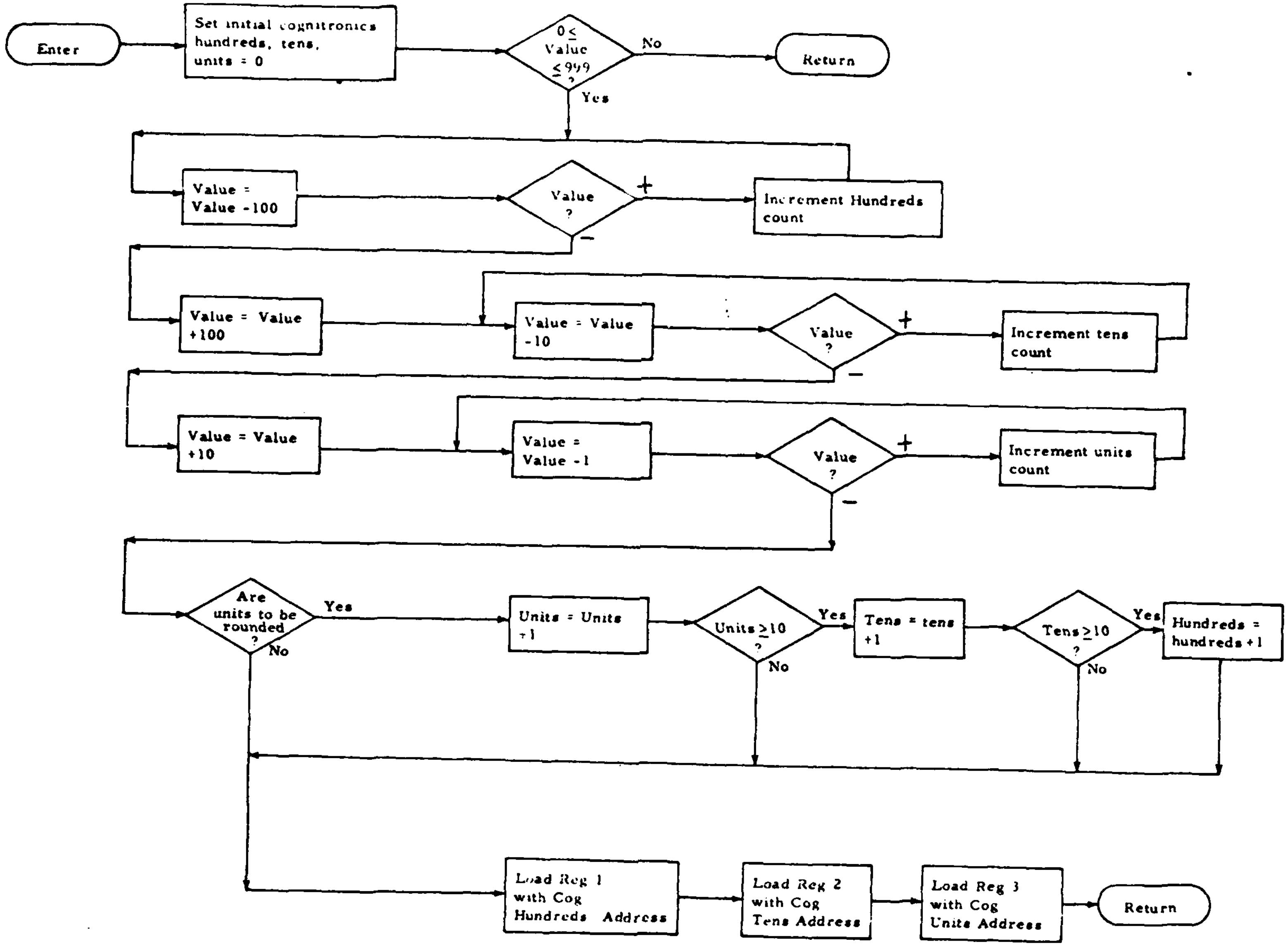


Figure 2-62. COGADDR Subroutine

2.28 AFT SUBROUTINES

a. Program Module Name. AFT Subroutines (ASUB)

b. Purpose. These subroutines are made available for use by various ATE/AFT program modules.

c. Requirements. The ASUB module is required to:

1. Provide trim values necessary to trim aircraft to a desired condition.

2. Compute the square root of any given number.

3. Compute ARCSIN (N) where N is a value between 0 and 1.

4. Convert integer values to floating point numbers.

5. Convert hexadecimal value to ASCII code for the IDIIOM display.

6. Convert floating point values to integer values.

d. Description. The ASUB module contains the following subroutines:

1. TRIM
2. SQRTO
3. ARCS
4. FLOAT
5. HEXASC
6. FIX

The TRIM subroutine is called by various modules in the AFT program to provide the necessary trim values needed to trim the aircraft to a designated flight condition. The TRIM subroutine will place the proper values for longitudinal, aileron, and rudder trim in locations accessible for use by the ATE$F module.

The SQRTO subroutine is called by the data processing module, IDP, when the square root of a number is required for scoring student performance. The method employs a very fast first approximation, subsequently used in two iterations of Newton's method which are performed simultaneously. Letting IAPR be the first or initial approximation, the next approximation would be calculated using

$$APR1 = 1/2\ (IAPR + F/IAPR) \quad (1)$$

The next approximation would be

$$APR2 = 1/2\ (APR1 + F/APR1) \quad (2)$$

Substituting the first equation into the second results in

$$4 * APR2 = IAPR + F/IAPR + \frac{4F}{IAPR + F/IAPR} \quad (3)$$

where

IAPR = the first or initial approximation

APR1 = the second approximation

APR2 = the third approximation

F = the number of which the square root is desired, modified by masking in the special exponent X'3F'.

This is the result of two iterations. The original exponent is halved since, to obtain the square root of NUMBER, where

$$NUMBER = 16^{n} \times \text{fraction} \quad (4)$$

SQRT is calculated to be

$$SQRT = 16^{n/2} \times \sqrt{\text{fraction}} \quad (5)$$

If n is even, equation (5) is adequate. If n is odd, an additional step is required which consists of multiplying equation (5) by a factor of four, where

$$NUMBER = 16^{n-1} \times 16^{n+1} \times \text{fraction} \quad (4A)$$

and

$$SQRT = 16^{(n-1)/2} \times 16^{1/2} \times \sqrt{\text{fraction}} \quad (5A)$$

$$\text{NOTE: } 16^{1/2} = 4$$

ARCSINE (N) is obtained in the range $=3\pi/8 \le N \le 3\pi/8$ using the following polynomial.

$$\text{ARCSIN (N)} = N \left[A_0 + \frac{A_1}{B_1 - \left(N^2 + \frac{A_2}{B_2 - N^2} \right)} \right]$$

where

$A_0 = 0.524998$

$A_1 = 1.578343$

$A_2 = 0.332159$

$B_1 = 3.557434$

$B_2 = 1.415690$

If ARCSIN (N) is outside the aforementioned range, the value sin (X) = cos (90-X) is used.

The FLOAT subroutine converts an integer value to a floating point number.

The HEXASC subroutine converts a hexadecimal value, through a table look-up procedure, to the appropriate ASCII character.

The FIX subroutine converts a floating point value to an integer value.

e. Inputs

1. Internal Inputs:

(a) TRIM subroutine. Calling location +1 contains the longitudinal trim value. Calling location +2 contains the aileron trim value. Calling location +3 contains the rudder trim value.

(b) SQRTO subroutine. The only internal input to SQRTO is the floating point number of which the square root will be computed, contained in register 8.

(c) ARCS subroutine. There are two internal inputs to ARCSINE(N)

(1) The sin ψ is in register 13 upon entry

(2) The cos ψ is in register 14 upon entry

NOTE: ψ is the angle to be determined in ARCSINE (N).

(d) FLOAT subroutine. Integer to be converted in R14.

(e) HEXASC subroutine. Enter with 4-byte hexadecimal word in R9.

(f) FIX subroutine. Enter with floating point number in R3.

2. External Inputs:

(a) TRIM subroutine. None.

(b) SQRTO subroutine. None.

(c) ARCS subroutine. None.

(d) FLOAT subroutine. None.

(e) HEXASC subroutine. None.

(f) FIX subroutine. None.

3. Constants:

(a) TRIM subroutine. None.

(b) SQRTO subroutine. A and B are predefined tables for computing the initial approximation.

(c) ARCS subroutine. RADDEG converts radians to degrees.

(d) FLOAT subroutine. None.

(e) HEXASC subroutine. TAB1 is table of hexadecimal characters and TAB2 is a table of ASCII characters for use in the table look-up routine.

(f) FIX subroutine. None.

f. Outputs

1. Internal Outputs:

(a) TRIM subroutine.	LTRIM	designated longitudinal trim
	ATRIM	designated aileron trim
	RTRIM	designated rudder trim

(b) SQRTO subroutine. The square root of NUMBER will be in register 8 upon exit.

(c) ARCS subroutine. ARCSINE (N) in degrees will be in register 14 upon exit.

(d) FLOAT subroutine. Exit with floating point number in R14.

(e) HEXASC subroutine. Exit with 4-byte ASCII word in R9.

(f) FIX subroutine. The integer value will be in register 3 upon exit.

2. External Outputs: The subroutines have no external outputs.

g. Program Entrances

1. TRIM subroutine.

```
BAL, 15    TRIM
DATA       LTRIM
DATA       ATRIM
DATA       RTRIM
```

2. SQRTO subroutine. BAL, 15 SQRTO

3. ARCS subroutine. BAL, 15 ARCSIN

4. FLOAT subroutine. BAL, 15 FLOAT

5. HEXASC subroutine. BAL, 15 HEXASC

6. FIX subroutine. BAL, 15 FIX

h. Exits

1. TRIM subroutine. B *15 (calling location +4)

2. SQRTO subroutine. B *15 (calling location +1)

3. ARCS subroutine. B *ARCEX (calling location +1)

4. FLOAT subroutine. B *15 (calling location +1)

5. HEXASC subroutine. B *15 (calling location +1)

6. FIX subroutine. B *15 (calling location +1)

i. Subroutines Called. None

j. Memory Requirements

1. TRIM subroutine:

Instructions	13
Data	1

2. SQRTO subroutine:

Instructions	29
Data	24

3. ARCS subroutine:

Instructions	41
Data	58

4. FLOAT subroutine:

Instructions	12
Data	9

5. HEXASC subroutine:

Instructions	25
Data	120

6. FIX subroutine:

Instructions	14
Data	2

k. Type of Program Module. The ARCS subroutine may be called only from ATE/AFT Foreground programs. The other subroutines may be called from the ATE/AFT Background programs only.

l. Flow Charts. See figures 2-63 through 2-68

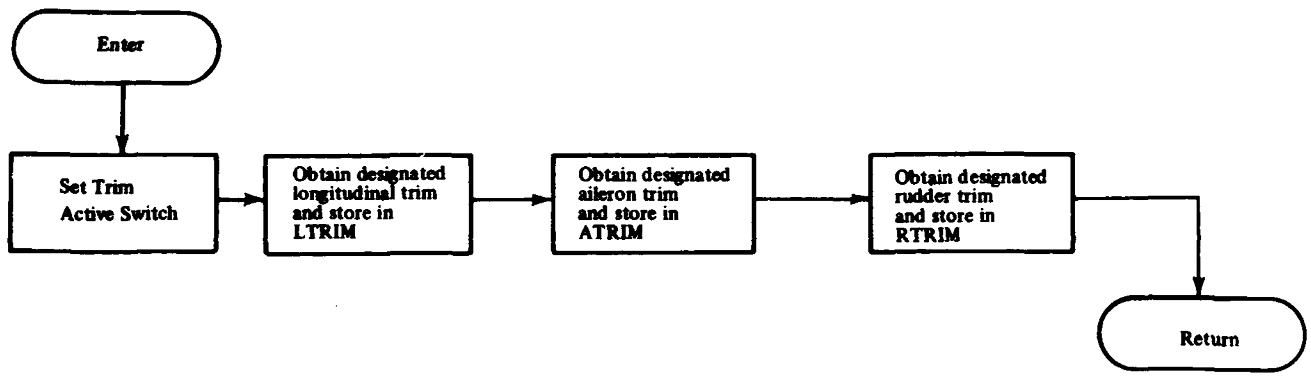


Figure 2-63. TRIM Subroutine

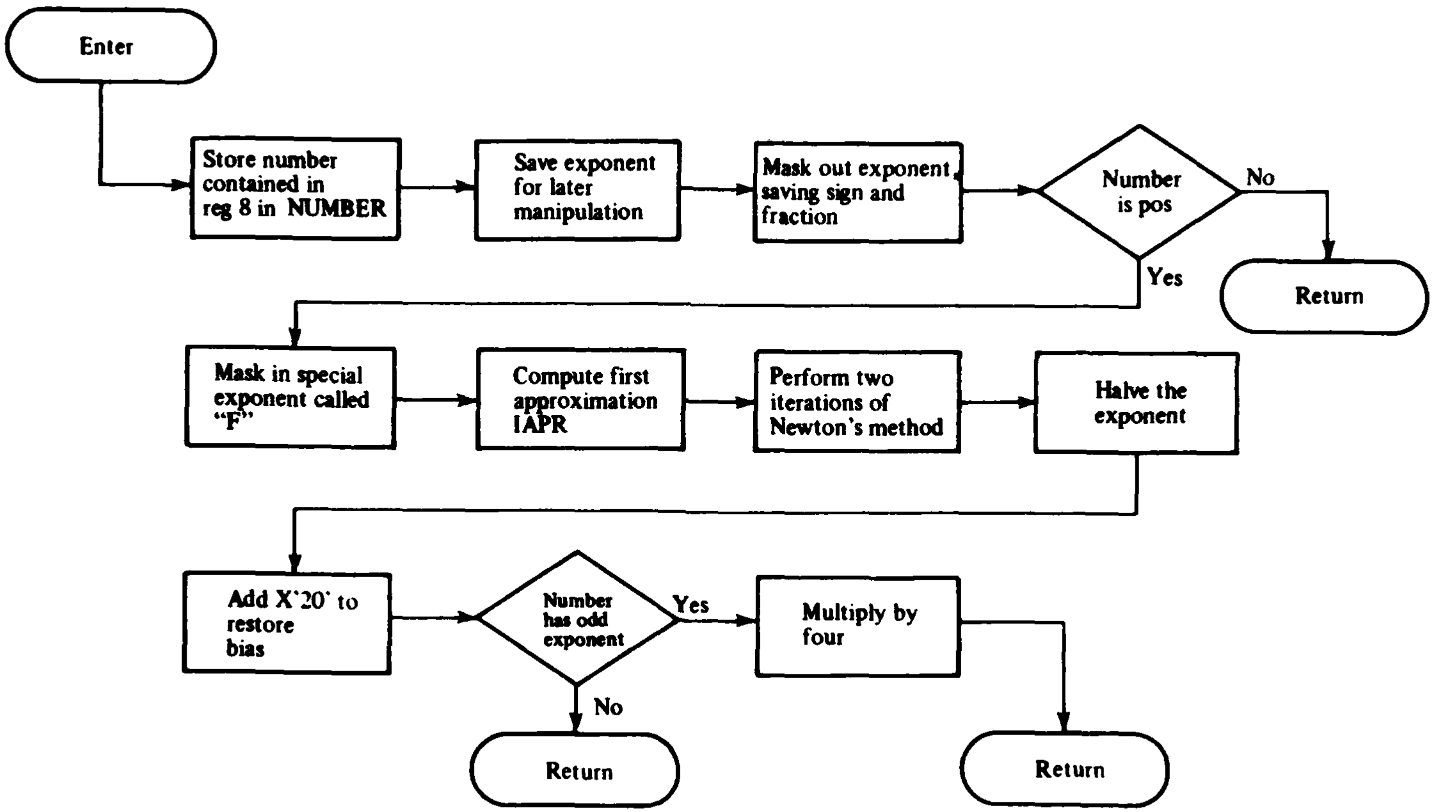


Figure 2-64. SQRTO Subroutine

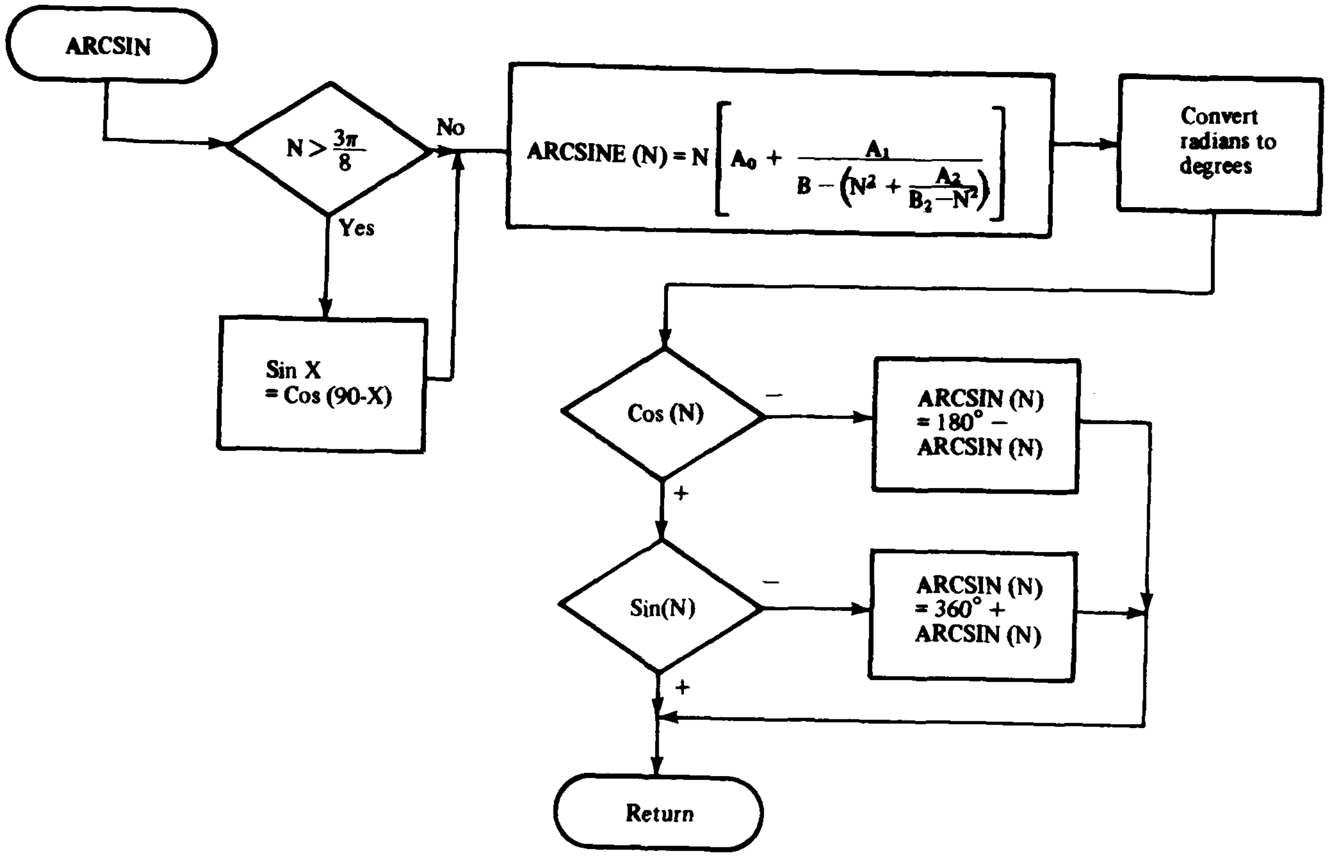


Figure 2-65. ARCSIN Subroutine

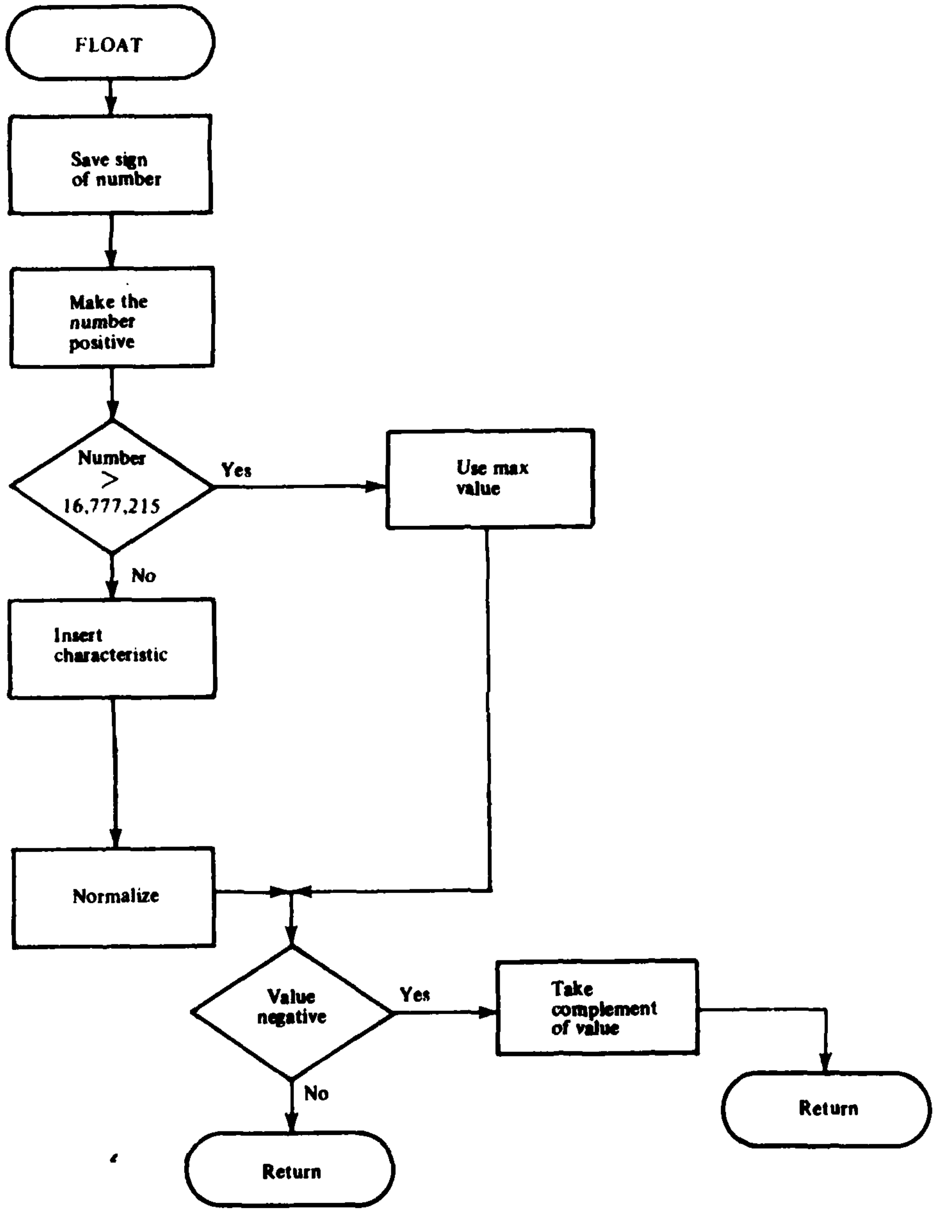


Figure 2-66. FLOAT Subroutine

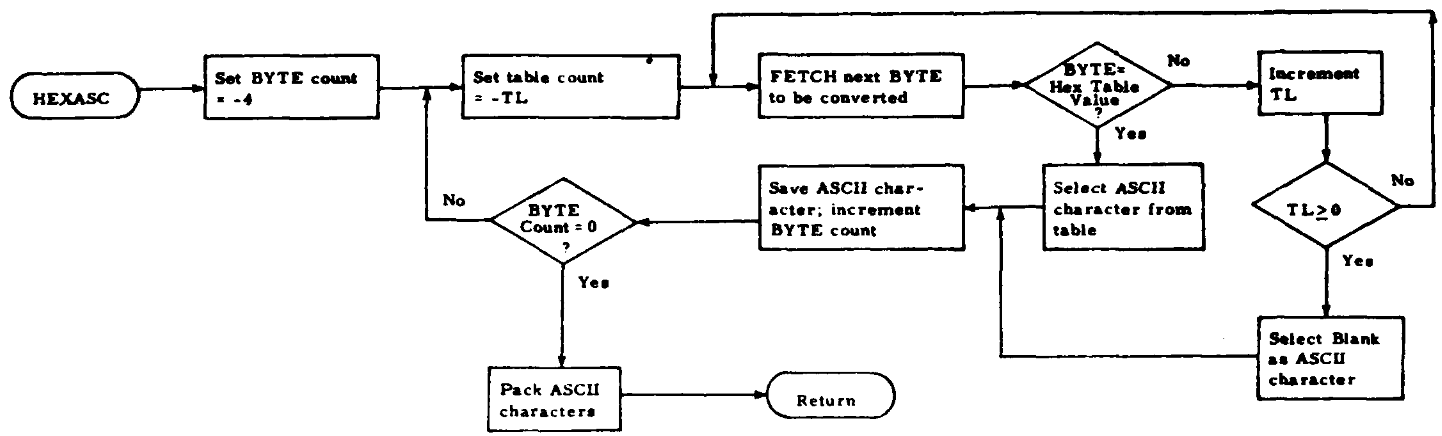


Figure 2-67. HEXASC Subroutine

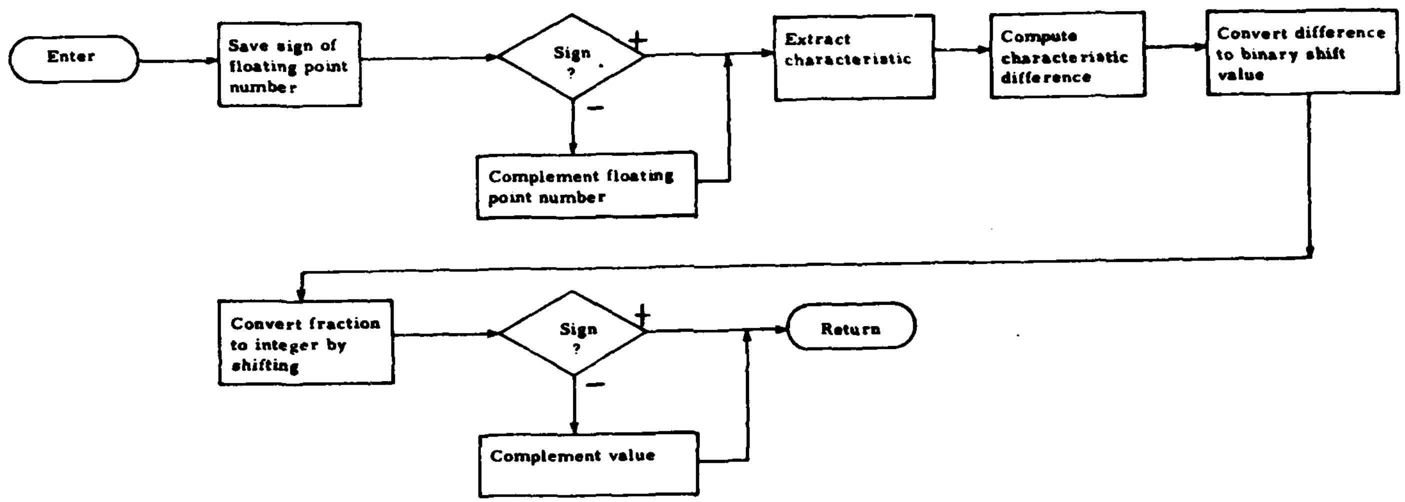
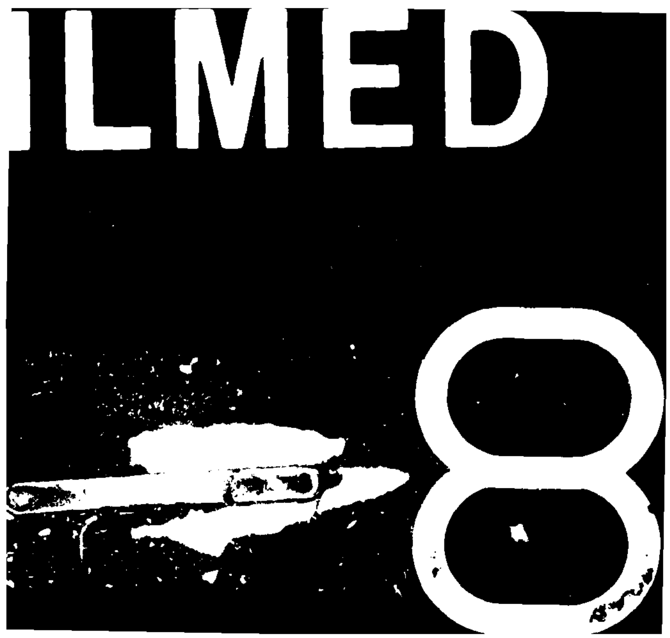


Figure 2-68. FIX Subroutine


ILMED
8